ASC-TR-95-1000

RAPID AERO-SHAPE GENERATOR (RAGE) - SBIR PHASE II

Mark D. Landon
Carrie Yeager

APTEK, Inc.
1257 Lake Plaza Drive
Colorado Springs, CO 80906-3578

6 August 1993

Final Report for Period July 1991 - August 1993

Approved for public release, distribution is unlimited.

AERONAUTICAL SYSTEMS CENTER

Air Force Materiel Command ▮ United States Air Force ▮ Eglin Air Force Base

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise as in any manner construed, as licensing the holder, or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This technical report has been reviewed and is approved for publication .

FOR THE COMMANDER

Daniel A. McInnis

Daniel A. McInnis
Chief, Weapons Branch
Studies and Analysis Division
Development Planning Directorate

Accesion For		
NTIS CRA&I		☒
DTIC TAB		☐
Unannounced		☐
Justification ________		
By ________		
Distribution /		
Availability Codes		
Dist	Avail and / or Special	
A-1		

Even though this report may contain special release rights held by the controlling office, please do not request copies from the Aeronautical Systems Center, Weapons Branch. If you qualify as a recipient, release approval will be obtained from the originating activity by DTIC. Address your request for additional copies to:

Defense Technical Information Center
Cameron Station
Alexandria VA 22304-6145

If your address has changed, if you wish to be removed from our mailing list, or if your organization no longer employs the addressee, please notify ASC/XREWS, 101 West Eglin Blvd., Suite 384, Eglin AFB FL 32542-5499, to help us maintain a current mailing list.

Do not return copies of this report unless contractual obligations or notice on a specific document requires that it be returned.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)

2. REPORT DATE
930806

3. REPORT TYPE AND DATES COVERED
Final Report 910715 through 930806

4. TITLE AND SUBTITLE
Rapid Aero-Shape Generator (RAGE) SBIR Phase II

5. FUNDING NUMBERS
573-3600-293-4769-
663005-256014-408E/409E-
65502F-527900

6. AUTHOR(S)
Landon, Mark D.
Yeager, Carrie

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
APTEK, Inc.
1257 Lake Plaza Drive
Colorado Springs, CO 80906-3578

8. PERFORMING ORGANIZATION REPORT NUMBER
F08635-91-C-0052

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Deputy for Development Plans
Aeronautical Systems Division
Eglin AFB, FL 32542-5000
Attn: Mr. G. Allen Baker - ASD/SRC

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
F08635-91-C-0052
ASC-TR-95-1000

11. SUPPLEMENTARY NOTES
N/A

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution Unlimited

12b. DISTRIBUTION CODE
A

13. ABSTRACT (Maximum 200 words)

The work completed in this Phase II SBIR developed and delivered two completely tested software packages. The first was RAGE (Rapid Aero-shape GEnerator) and the second was TrajView (real time 3-D trajectory viewer). Rage helps determine and display 3-D, optimal, geometric aero-shape designs and analysis results. Rage used Free-Form Deformation, optimization software (OptdesX), hypersonic/supersonic aerodynamic prediction software (HAB), and Moviestar for geometry and finite element modeling, computer graphics model visualization. TrajView displays and animates (in real time) the trajectories of multi-body or multi-trajectory predictions and their associated data. TrajView was written in Silicon Graphics IRIS Inventor.

14. SUBJECT TERMS
Shape Optimization, Hypersonic, Aerodynamics, Computer Graphics, Design Optimization, Trajectory Animation

15. NUMBER OF PAGES
75

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std Z39-18
298-102

Table of Contents

Introduction

This report discusses the work completed under Air Force SBIR contract F08635-91-C-0052 titled, "Rapid Aero-shape GEnerator (RAGE) - SBIR Phase II" by APTEK, Inc. This report contains a technical description and a user's manual/tutorial with test case examples for the Rage Shape Optimization software and the TrajView Trajectory Animation software developed under this contract. Phase III considerations are also discussed in this report.

The overall objective for this Phase II SBIR work was to complete the development of the Phase I work and to deliver two computer software programs. One program is Rage, Rapid Aero-shape GEnerator, and the other program is TrajView, real time trajectory animation software for the visualization of 6 degree-of-freedom trajectories and associated data. These programs contain the capabilities to help government personnel determine and display 3-D, optimal, geometric aero-shape designs, analysis results, and trajectory information. The software includes:

- X-Windows Motif graphical user interface.
- Improved 3-D geometry and finite element modeling.
- Improved computer graphic display.
- General purpose shape optimization capability.
- Improved Free-Form Deformation methodology.
- Improved graphical display of missile trajectory animations.
- Improved design optimization software.
- Additional geometry/finite element translation software.
- Etc.

The specific objectives for this Phase II work are listed here.

1. Enhance the aero-shape optimization software. One objective here was to enhance the shape optimization software to generate a family of aero-shapes with control fins to have identical aerodynamic coefficients. Another objective here was to add the capability for a model's wings and control fins to have hinge lines for rotation. Also, it was important to provide the capability to find shapes that matched trimmed aerodynamic coefficients via the shape optimization process.

2. Develop uniform shrink algorithms. One objective here was to develop an algorithm that derive the solid geometry definition of uniformly thick shells. The other objective here was to develop an algorithm to derive the receding geometry and mass properties for a solid rocket motor burn simulation.

3. Develop three-dimensional (6 degree-of-freedom) trajectory plotting capabilities. The objective here was to provide capabilities to easily display 3-D trajectories of missiles, aircraft and targets, and related data with color computer graphics. This included 1) a model of the globe earth, 2) a flat earth model with Cartesian grid, 3) ground trace and flight path trace, 4) altitude hacks, 5) time ticks, 6) elliptical tube for display of INS errors, 7) odometer type displays for associated data, 8) real time animation, 9) multiple trajectory displays, and 10) text display.

4. Implement the surface intersection algorithm into Moviestar. The objective here was to include the geometric hermite intersection algorithm into Moviestar. Another objective here was to improve Moviestar's line and surface definitions to be based on the non-uniform rational b-spline (NURBS) to lay the foundation for the non-Coon's patch boundary representation of solids.

5. Investigate and develop a database translator between the ASD/XRHI database (from Wright-Patterson) to the Moviestar database. The objective here was to interface the ASD/XRHI database to Moviestar thus allowing Moviestar to be a pre- and post-processor for XRHI database models.

6. Convert the F-16 model in the ASE database format to a Moviestar database format. The objective here was to provide a Moviestar geometry model of the F-16 aircraft for future physical fit compatibility studies.

7. Hardware and software compatibility. The objective here was to insure that all software developed will operate on Eglin AFB computers.

8. Provide documentation, training and periodic reporting. The objective here was to deliver the completed software, to train government personnel in its operation, and to provide documentation of progress throughout the duration of the contract.

Each objective above was achieved by completing the tasks discussed below. The following Technical Description contains a detailed report on the work completed in each task. There are several figures showing the results of the work, however, the reader will periodically be referred to the user's manuals and tutorials in Appendix A and B. The user's manuals and tutorials contain a lot of examples and test cases that show what the user would see on the screen of a workstation while executing the software.

This work was completed in cooperation with another Phase II SBIR with the Naval Surface Warfare Center (NSWC) in Silver Spring, MD (contract number N60921-90-C-0280 titled, "An Optimal Maneuvering Reentry Body Shape Design Package - SBIR Phase II")

Technical Description

The work under this Air Force Phase II contract was carried out by completing tasks in eight main areas. The work in several of the tasks incorporated, modified and linked existing computer graphic, design optimization and computer aided geometry tools together to produce the final software programs. The specific tools for Rage include Free-Form Deformation (FFD), hypersonic aerodynamics (HAB, also known as SHABP), design optimization (OptdesX) and solid modeling, finite element generation, and computer graphics (Moviestar.BYU). The specific tools for TrajView include Inventor which is a Silicon Graphics programming environment that provides basic tools for 3-D real time animation on a Silicon Graphics Workstation. Each of these are briefly reviewed here.

Free-Form Deformation.

FFD is the process for the 'molding' of an initial shape definition into a different shape. FFD is defined in terms of a tensor product trivariate Bernstein polynomial. A local coordinate system is imposed on a parallelepiped region as shown in Figure 1a. Any point $\mathbf{x}$ has (s,t,u) coordinates in this system such that

$$\mathbf{X} = \mathbf{X}_0 + s\mathbf{S} + t\mathbf{T} + u\mathbf{U}. \quad (1)$$

The (s,t,u) coordinates of $\mathbf{X}$ can easily be found using linear algebra. A vector solution is

$$s = \frac{\mathbf{T}\times\mathbf{U}(\mathbf{X})-\mathbf{X}_0}{\mathbf{T}\times\mathbf{U}\cdot\mathbf{S}}, t = \frac{\mathbf{S}\times\mathbf{U}(\mathbf{X})-\mathbf{X}_0}{\mathbf{S}\times\mathbf{U}\cdot\mathbf{T}}, u = \frac{\mathbf{S}\times\mathbf{T}(\mathbf{X})-\mathbf{X}_0}{\mathbf{S}\times\mathbf{T}\cdot\mathbf{U}}.$$

Note that for any point interior to the parallelepiped that $0 < s < 1, 0 < t < 1, 0 < u < 1$.

Next, a grid of control points $\mathbf{P}_{ijk}$ on the parallelepiped is imposed. These form $l+1$ planes in the $\mathbf{S}$ direction, $m+1$ planes in the $\mathbf{T}$ direction, $n+1$ planes in the $\mathbf{U}$ direction. In Figure 1b, $l = 1$, $m = 2$, $n = 3$. The control points are the corner points of the FFD grid. In Figure 1b, $l = 2$, $m = 4$, $n = 2$, and $(l+1)(m+1)(n+1) = 45$ control points. These points lie on a lattice, and their locations are defined

$$\mathbf{P}_{ijk} = \mathbf{X}_0 + \frac{i}{l}\mathbf{S} + \frac{j}{m}\mathbf{T} + \frac{k}{n}\mathbf{U}.$$

The deformation is specified by moving the $\mathbf{P}_{ijk}$ from their undisplaced, latticial positions. The deformation function is defined by a trivariate tensor product Bernstein polynomial. The deformed position $\mathbf{X}_{ffd}$ of an arbitrary point $\mathbf{X}$ is found by first computing its (s,t,u) coordinates from equation (1), and then evaluating the vector valued trivariate Bernstein polynomial:

$$\mathbf{X}_{ffd} = \sum_{i=0}^{l} \binom{l}{i} (1-s)^{l-i} s^i \left[\sum_{j=0}^{m} \binom{m}{j} (1-t)^{m-j} t^j \left[\sum_{k=0}^{n} \binom{n}{k} (1-u)^{n-k} u^k \mathbf{P}_{ijk} \right] \right] \quad (2)$$

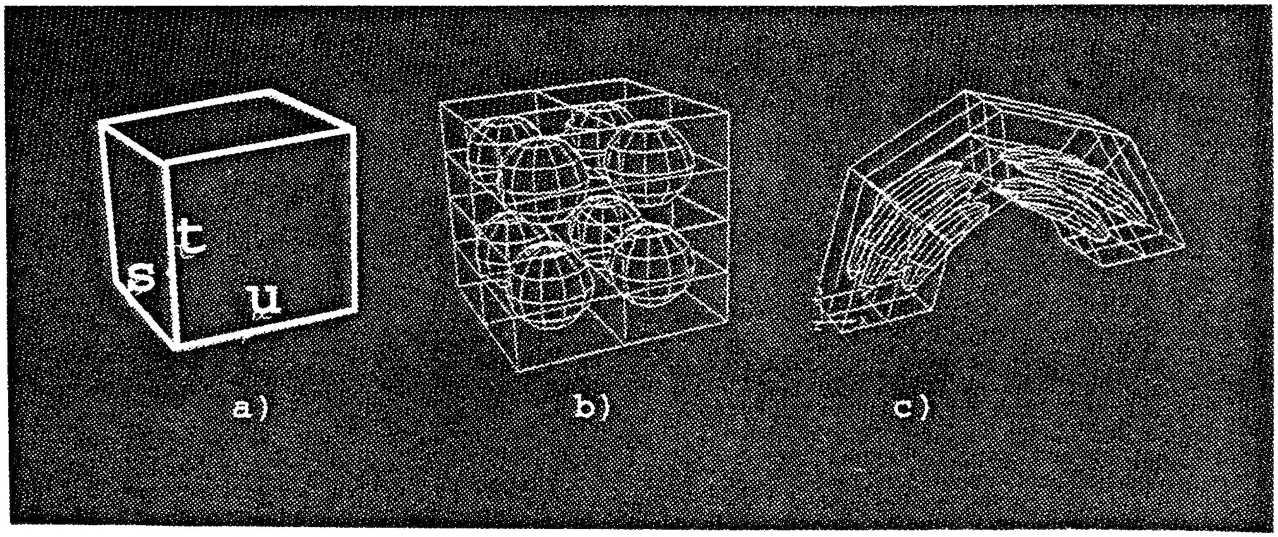


Figure 1: a) FFD local coordinate system, b) 45 control point lattice, c) displaced control points and resulting deformation of polygonal model.

where $\mathbf{X}_{ffd}$ is a vector containing the Cartesian coordinates of the displaced point, and where $\mathbf{P}_{ijk}$ is a vector containing the Cartesian coordinates of the control point. Figure 1c shows the displaced control points and the resulting displaced pattern of the spheres that were inside of the initial parallelepiped.

FFD allows great flexibility in shape definition while it reduces the number of design variables necessary to arrive at optimal shapes. The user may move the control points to deform the model (that is inside the control point grid) and 'mold' it like clay in the hands of a sculptor. In the shape optimization application the optimization module moves the control points to deform the model to its optimal shape.

SHABP: The Hypersonic Aerodynamic Analysis Software.

The HAB code was used for the aerodynamic coefficient calculations. HAB is a modified version of the MARK IV Supersonic/Hypersonic Arbitrary Body Program. HAB is capable of calculating the hypersonic (and supersonic) aerodynamic characteristics (coefficients) of complex arbitrarily shaped 3-D bodies. The NSWC Phase II SBIR funded work that modified the HAB code to 1) calculate more accurate viscous results, 2) predict the trajectory dependent ablation of leading edges of wings and noses, and 3) calculate the maximum lift-to-drag ratio (as a function of the angle of attack) for each analysis call in the optimization process. This Air Force Phase II funded improvement to HAB to calculate the trimmed aero-coefficients, at each angle of attack defined by the user, to allow for aero-coefficient matching with trimmed aero-coefficient curves.

OptdesX: Design Optimization.

OptdesX, a commercial optimization software package from Design Synthesis Inc., was used for the design optimization tasks. It has several robust and efficient mathematical optimization search routines that help find the values of design variables defining optimal design solutions. The graphical user interface for OptdesX was developed using X-windows Motif on Unix machines. This Phase II contract funded the implementation of OptdesX for use on DEC (VMS) X-windows Motif workstations.

The NSWC Phase II SBIR work added a Simulated Annealing search routine that helps find global optimal solutions.

Moviestar: Geometry/Finite Element Generation And Computer Graphics.

Moviestar is a geometric modeling program that allows the user to define points, curves, surfaces and solids that define the geometry of aero-shapes. The user may define the shapes that are used for the creation of finite elements (surface facets) that are then input into the HAB code for an aerodynamic analysis. Moviestar's user interface facilitates data input, model manipulation, automatic mesh generation, computer graphic visualization, data base translation, etc..

Inventor: Silicon Graphics 3-D Graphics Toolkit.

Inventor is an object-oriented 3-D toolkit. It contains a set of building blocks that enable the programmer to write programs that take advantage of the powerful 3-D graphics hardware features on a Silicon Graphics Workstation with minimal effort. It provides a library of objects that the programmer uses and modifies to meet new needs. It is based on the Silicon Graphics Library (GL) which allows for 3-D objects to be moved (translated and rotated) in real time in the scene viewer on the screen.

A discussion of the results obtained under each task is reported here.

TASK 1: Aerodynamic Shape Optimization for Aero-Coefficient Matching.

Aptek modified and enhanced the Phase I Rage Shape Optimization software to include the following items. The use of these capabilities are shown in the user's manual in Appendix A.

- All aero-coefficients from HAB are available for use as analysis functions in the optimization process. These are the axial, normal, drag, lift, and side forces, the pitch, yaw and roll moments, and the lift to drag ratio for a user defined angle of attack and for the angle of attack at the maximum lift-to-drag ratio value. Other analysis functions are now available such as, the gradients of the pitch moment and normal forces with respect to the angle of attack, the gradients of the yaw moment and side force with respect to slip angle, the aero-centers for pitch and yaw, and mass properties (xcg, ycg, Ixx, Iyy, etc..)

- The capability to allow the user to input target trimmed aero-coefficient curves (for the drag coefficient and the lift-to-drag as functions of the angle of attack) with a user definable number of points along the angle of attack was incorporated into the initialization window.

- The user is allowed to put a separate FFD grid around each group of user definable panels. This includes the capability to put an FFD grid around each fin also.

- The user is allowed to define a hinge line for each fin by defining a point and vector about which the fin is rotated. The fin may be translated with X, Y, and Z translation as analysis variables also.

- The fin to body interface can be forced to remain intact by defining a body FFD control point that the fin will translate with during the deformation of the bodies FFD grid.

- The capability to calculate the trimmed aero-coefficient curves for each analysis call was developed. The subroutine called HABTrim was developed and is called in the ANAFUN subroutine (i.e. each analysis call in the optimization process). HABTrim is given the current shape of the aero-shape along with each angle of attack defined in the target trim curve definition. Then it minimizes the absolute value of the pitching moment to zero using the control fin's rotation as a variable. If the pitching moment cannot be forced to zero, then the configuration is not able to be trimmed with the current fins size and/or location. A warning message is printed if this is the case. The aero-coefficients calculated this way define the predicted trimmed aero-coefficient curves. Two additional analysis functions are made available to be used in trimmed aero-coefficient matching. These are the summation of the least square differences between the target trimmed curves and the predicted trimmed curves for both the lift-to-drag ratio and the drag coefficient. The user may then define an objective function to minimize the least square differences. As OptdesX performs the shape optimization in this case, the shape is modified so that its trimmed aero-coefficient curves match the targeted trimmed aero-coefficient curves.

- The capability to model the initial aero-shape to be conformal to another shape (i.e. an aircraft wing) is available in Moviestar. The capability to keep the conformal surface's initial shape the same is available by allowing the user to define

a list of nodes on the model to remain immune from FFD deformation. The window that allows for the target trimmed curve definition also has a place to define the number of immune node as well as the actual node number to remain immune from deformation. This capability will allow the user to design conformal weapons will optimizing the shape with desirable aero-coefficients.

- The X-windows Motif graphical user interface was originally developed on Unix systems. It was modified to run under the DEC (VMS) operating system.

TASK 2: Algorithm Development for the Uniform Shrink of a Solid.

A general surface offset algorithm was developed to define an offset surface where the distance between the original surface and the offset surface (everywhere normal) is a uniform thickness. This algorithm is able to be used in two ways. One way is to allow the user to model the shape of a missile (i.e. nose and body) and create a solid shell thickness by offsetting the surfaces that define the shape. The new offset surfaces and the original surfaces bound the solid shell volume. The second way to use this algorithm is to simulate the change in mass properties of a rocket motor burn sequence. The user here performs the offset of the burn surfaces repeatedly (in small increments) and recalculates the mass properties for each increment. An example of the later is shown in Figure 2. Each capability has been implemented in Moviestar by adding user friendly commands in the Moviestar graphical user interface.

TASK 3: 3-D Trajectory Plotting Capability.

Aptek developed a complete software package called TrajView to display and animate 3-D trajectories calculated with the program CADAC at Eglin AFB. TrajView was developed on Silicon Graphics Workstations with the IRIS Inventor programming library. This was necessary because of the high powered graphics library (GL) that Silicon Graphics Workstations have. Real time 3-D translations and orientations are available with this library of graphics calls.

Samples of what the user sees while running TrajView (such as the user interface windows and graphical displays) are shown in the tutorial and user's manual (Appendix B). Many test cases were executed with TrajView to insure its robustness. TrajView was developed to include all of the following capabilities.

- TrajView reads CADAC data directly from disk files into its own data records. This is facilitated by a user friendly window that pops up and the user simply clicks the mouse button on the desired file of the files listed.

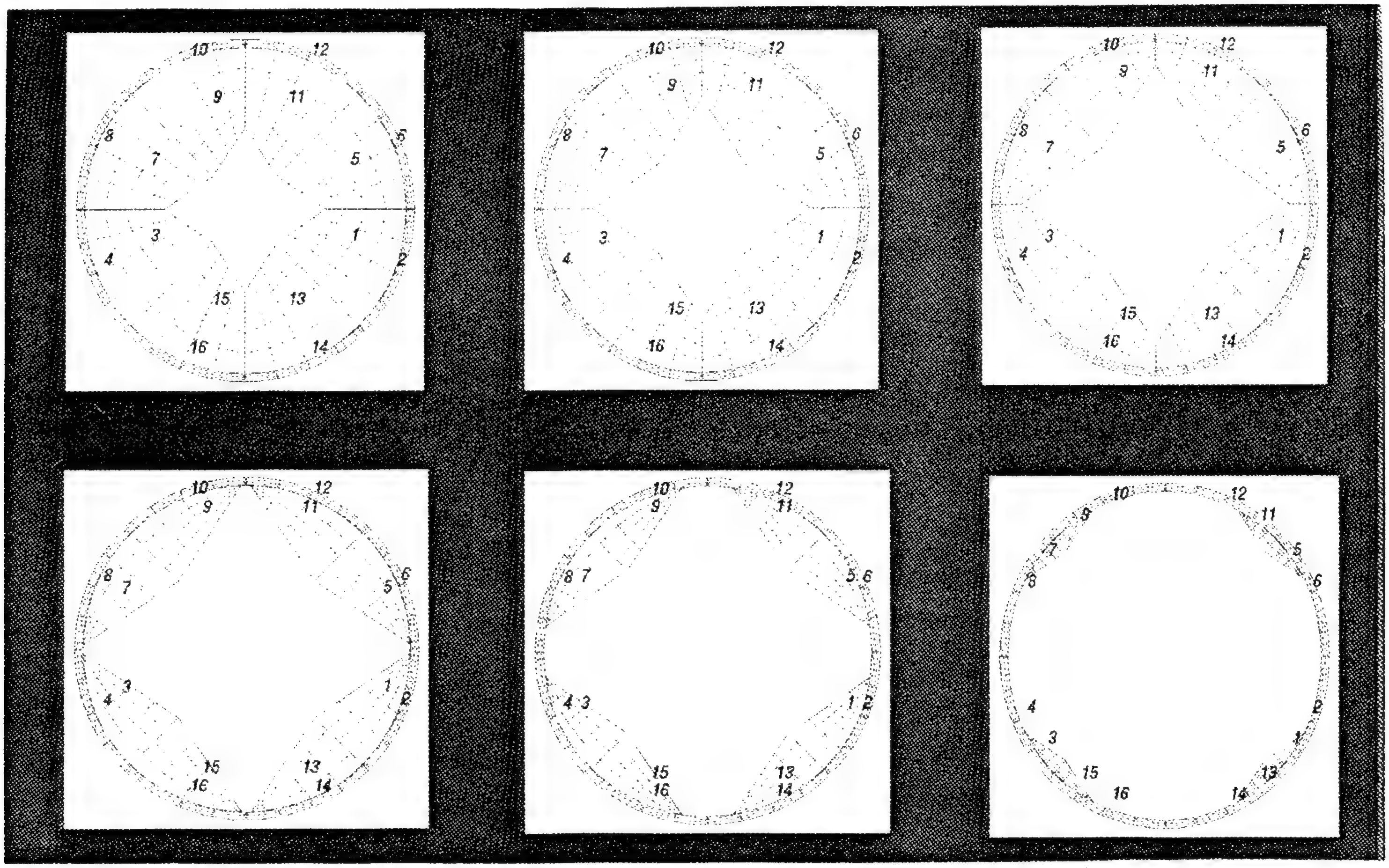


Figure 2: A star shaped rocket motor burn simulation in six equal increments

- Multi-body trajectories or single body multi-trajectories may be displayed. The multi-trajectory displays all trajectories of one body simultaneously. The user may optionally delete any of the multi-trajectory displays.
- A globe earth or a flat earth model may be displayed.
- The flat earth model shows a Cartesian grid of down range, cross range, and altitude.
- Optional ground and flight traces are displayed along with flight trace projections on the side and back walls.
- Optional vertical stringers with altitude hacks are displayed at constant time intervals.
- A transparent tube is displayed showing the INS errors in the CADAC data file.
- The user may define any number of CADAC variables to be displayed (in a separate window) with odometer/slider type displays. The slider bar moves back and forth in the graphical viewing window, as the animation proceeds, showing the current values as a function of the model's time.
- The user may generate static 2-D plots of any CADAC variable versus any other CADAC variable as well. These plots may be saved with a hardcopy command for incorporation in reports.

- The ability to animate 3-D objects was developed with the user having control over the speed of the animation (i.e. real time, fast/slow forward and reverse, or at many speeds forward and reverse). A display of the model's time and the clock time are given side-by-side to allow the user to verify whether or not the animation is real time. Depending on the memory and graphics engines on the workstation being used, the animation may be slower than the rate of time chosen for the animation. The animation window simulates a VCR type display with play, reverse, fast forward, fast reverse, stop, and a slider bar to vary the animation speed as well.

- Simple 3-D models were developed to be used in the animations. An F-15 and an F-16 were modeled for aircraft animations. The JDAM missile was modeled for missile animations. A tank and a bridge were modeled for targets. A unit vector was modeled to be used for velocity vector animation displays. Two separate files for each model are given for both the globe earth and the flat earth initial orientations. Figure 3 shows several of the models created for the trajectory animations.

- Hardcopies of any of the screen displays are available with the Silicon Graphics screen capture tool. These images can be included in reports or printed directly to a postscript printer.

- Text can be superimposed on any display if the display is captured and imported into Silicon Graphic's Showcase program. Text, lines, arrows, other captured displays, etc. can be included to make attractive documents and reports. The user can choose the font, size, color, and style of the text input into the document. The tutorials in this report were created with Showcase.

TASK 4: Surface Intersection Algorithm and NURBS in MOVIESTAR.

Moviestar was enhanced to allow the user to define NURBS (non-uniform rational B-splines) entities. The database was expanded to allow for storage of control points, curves, surfaces and solids as NURBS. The user interface was also modified so there is a separate menu for just NURB geometry input. NURB surfaces and solids may be meshed and grouped just like ordinary Moviestar surfaces and solids. However, in defining a solid, we cannot mix NURBS and Coon's Patches. The new NURB entities have been tested with the intersection algorithm, the meshing commands, the grouping commands (including the scale, translate, rotate, quaternion rotate, etc...). All NURBS entities can be used in Rage.

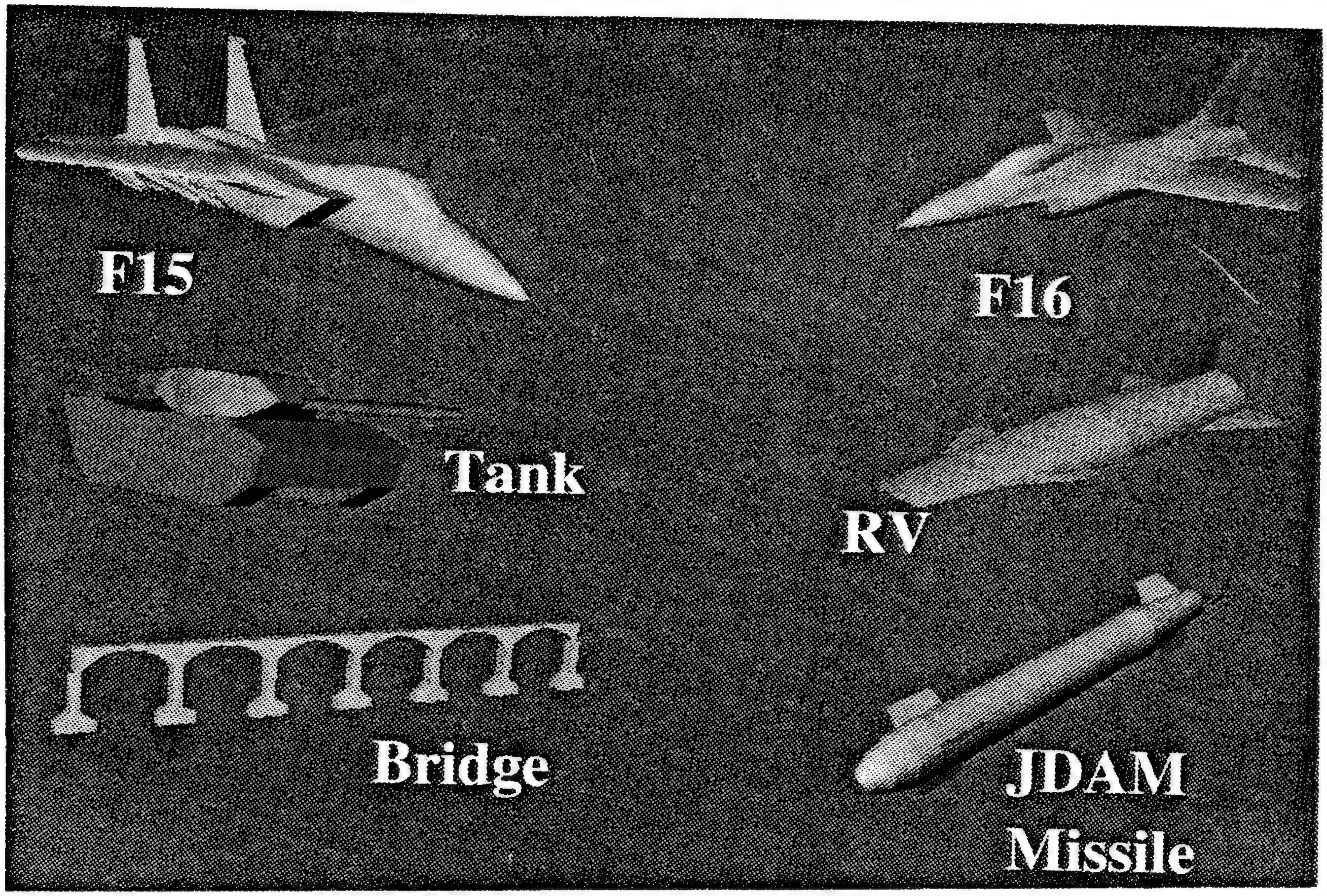


Figure 3: Several of the models created for TrajView animations.

TASK 5: Database Translator Between Wright-Patterson and Eglin Databases.

A reverse translator (from Write-Patterson's ASD/XRHI to Moviestar) was developed. This translator takes the hoops and stringers information in the ASD/XRHI format and converts them into finite elements and nodes. For each node a point is defined and these points can then be used by the user to create curves, surfaces and solids by hand. This translator was tested with several files containing ASD/XRHI models.

TASK 6: Convert ASE F-16 Database Into MOVIESTAR Solid Model.

Aptek converted an ASE F-16 database into a Moviestar solid model. The process consisted of using the existing ASE to Moviestar translator to convert the facets into Moviestar finite elements, nodes and points. The points were connected into curves, the curves into surfaces and solids. Figure 4 shows several of the different components of the F-16 model as Moviestar solid entities.

Aptek created a Moviestar solid model of the TER bomb ejection rack. Figure 5 shows all of the solids that make up the TER solid model.

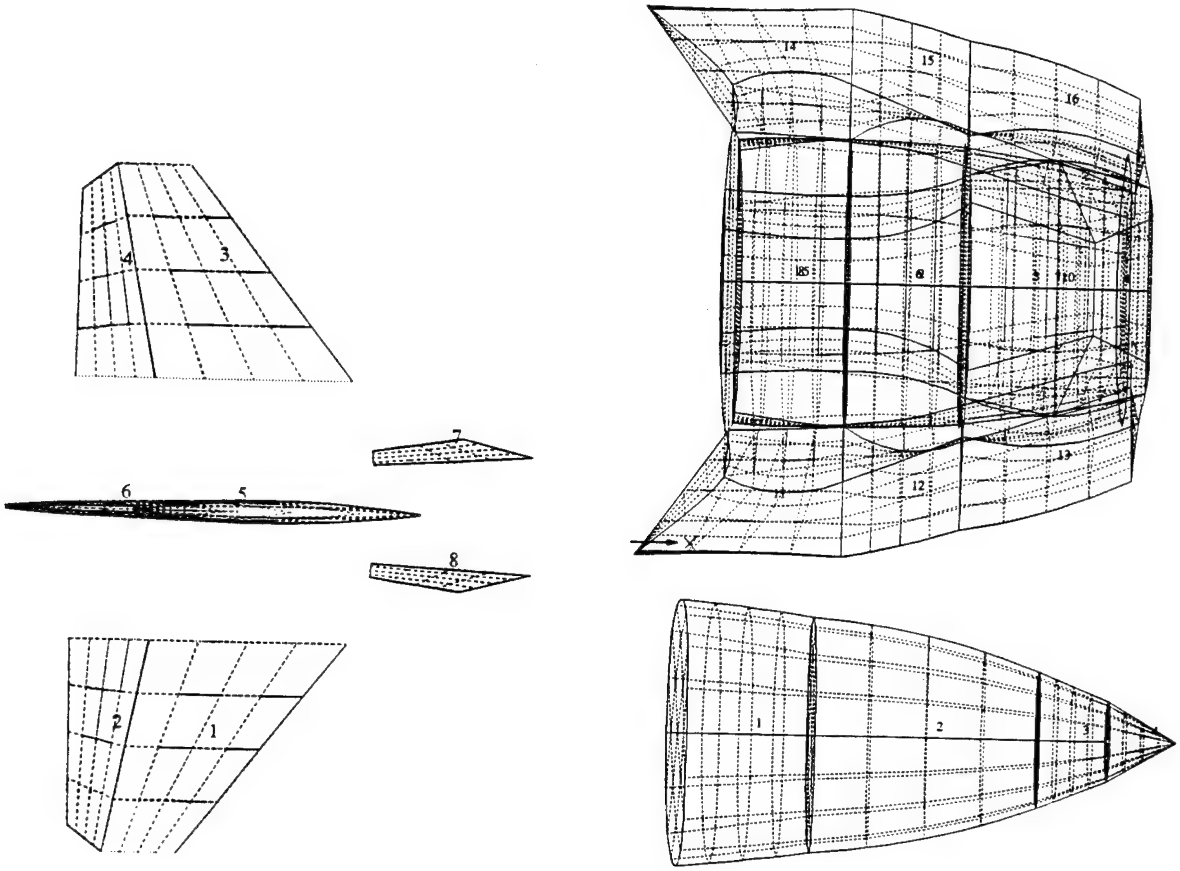


Figure 4: Moviestar solid entities of the F-16 aircraft.

Figure 5: Moviestar TER solid model.

Task 7: Hardware Compatibility.

The software was developed to be as machine portable as possible. Rage, Moviestar, and OptdesX can be run on any workstation that runs X-windows Motif.

Because of Inventor's dependency on Silicon Graphics Graphics Library, TrajView can only run on Silicon Graphics Workstations for now. However, Open GL will soon be available on many other workstations, such as, DEC, Hewlett Packard, PCs (with NT), etc.

Task 8: Documentation, Training and Reporting.

Periodic progress reports were written to communicate current progress on all tasks throughout the duration of the contract.

Two midterm reviews were held. One was held at Eglin AFB and the other was held at Aptek in Colorado Springs.

A final review and briefing was held and included software installation and a two day training session for the Rage and TrajView software packages. All software necessary to run Rage and TrajView was installed, including Moviestar, OptdesX (object code only), Rage, and TrajView. All run time help files were also installed.

Sufficient copies of all reports, briefings, and user's manuals were delivered.

Phase III Considerations.

APTEK is committed to the further development and marketing of the exciting capabilities that were investigated and developed under this Phase II effort. The Phase II work greatly enhanced the possibility of Phase III work and marketability. We feel that the Rage and TrajView software is at a point that should be attractive to government and commercial entities alike. We have created several high quality brochures showing what these software packages can do. We have attended Small Business Seminars and Conferences to market our capabilities with modest results.

Several of the areas where we feel Phase III opportunities exist are:

- Continued research and development in the shape optimization and data visualization areas for use in the government such as NASA, Navy, Army and of course the Air Force. Other agencies could request the inclusion of other analysis routines that would provide other design criteria such as radar cross section signature reduction for stealth aircraft, tanks and weapons, computational fluid dynamics for

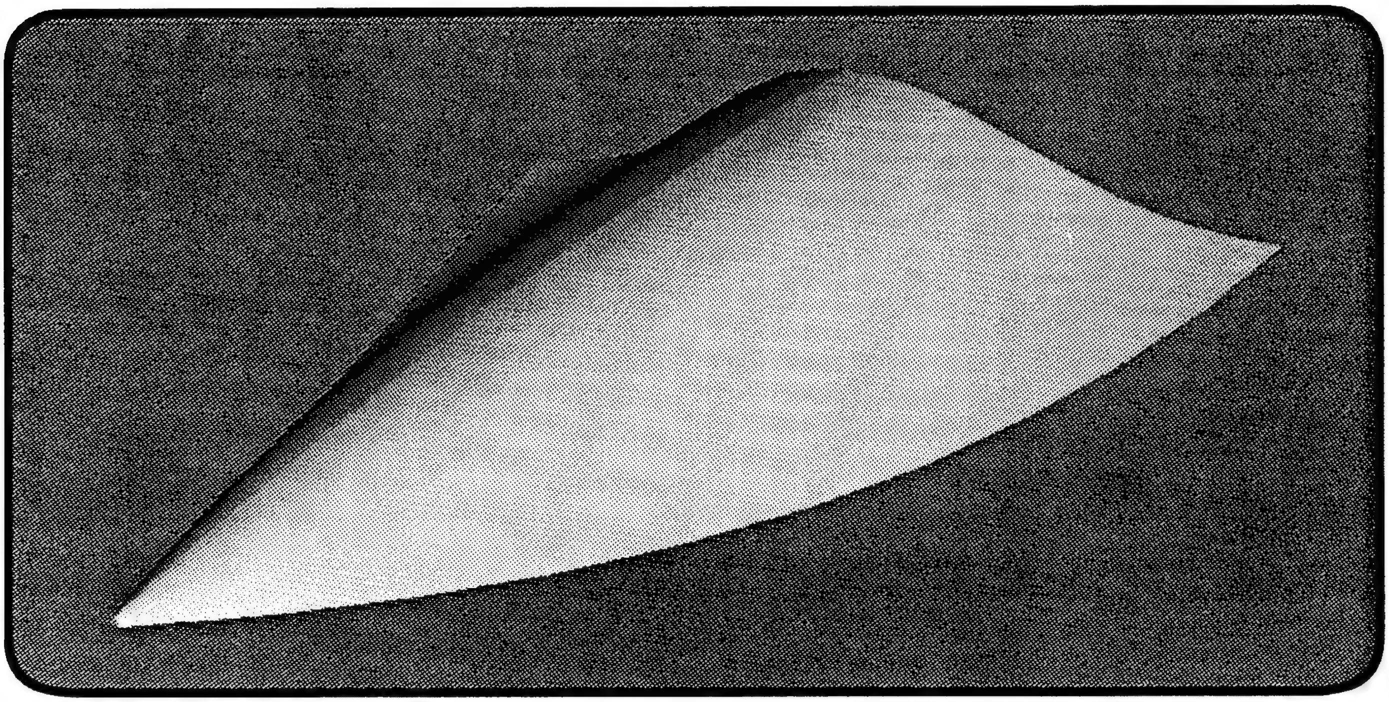
optimal shape and flow around submarines, space craft, weapons, etc., structural shape optimization, acoustic analysis for sound's interaction with shapes, waterways and costal areas for optimal shapes according to erosion and flood control, etc.

- Continued research and development in the shape optimization and data visualization areas for use by commercial customers. Commercial firms could request the inclusion of other analysis routines that would provide design criteria such as computational fluid dynamics for optimal shape and flow around automobiles, space craft, etc.,
- Perform software sales and training for commercial users.
- Complete specific problem solving for government and commercial users.

Appendix A

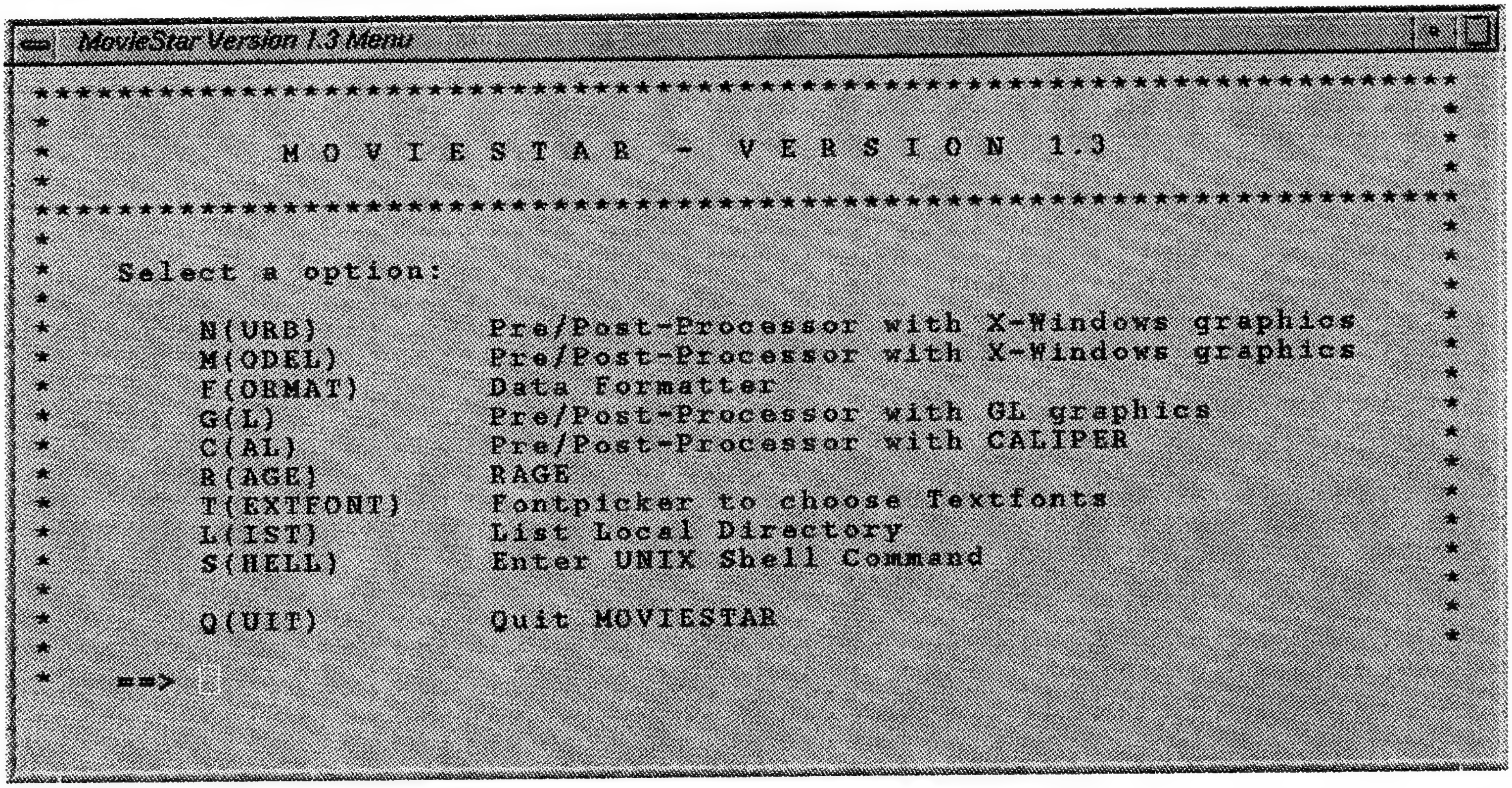
RAGE User's Manual And Tutorial

RAGE Tutorial



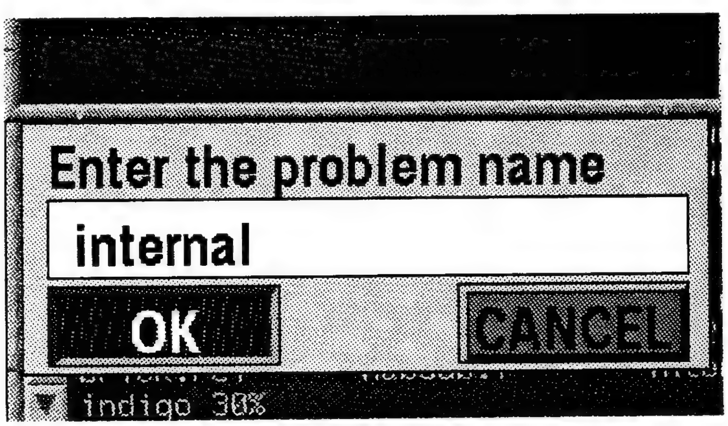
This tutorial will show the user the necessary steps to take to use the RAGE software package for aero–shape optimization. The various tools and functions available in Rage will help the user find optimal aero–shapes. First, Moviestar is used to model the initial geometry, mesh the surfaces, and group the meshed surfaces into panel groups. Second, Rage is executed and the Aero/FFD initialization window is filled to define the problem's desired aerodynamic input data, free–form deformation control, each group's mass properties, immune geometry, trimmed aero–coefficients, etc... Third, Rage (with Optdesx) is continued to setup the desired optimization variables, objective function and constraints. And finally, other tools are available to explore design space, view the models design evolution, modify problem, ect...

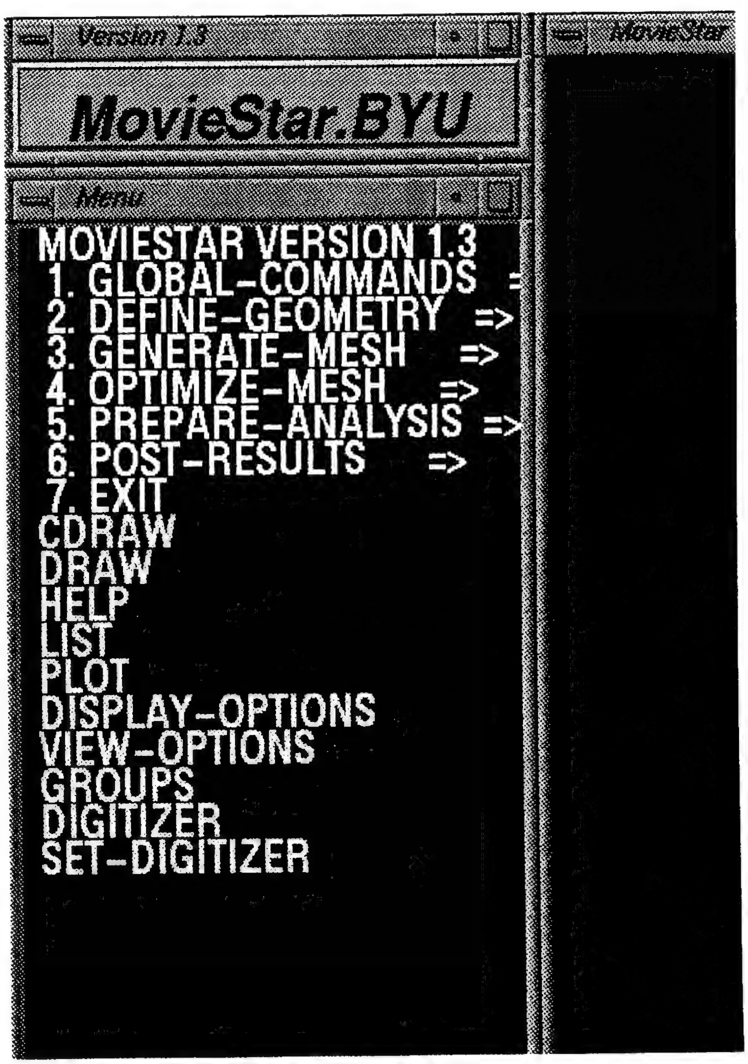
Step 1: In a console window, type **mvstart** and this window will appear.



Step 2: Type: **c** to start **Moviestar.**

A request for the model's name will appear (below). After typing in the model's name, the **Moviestar** menu will appear (right).

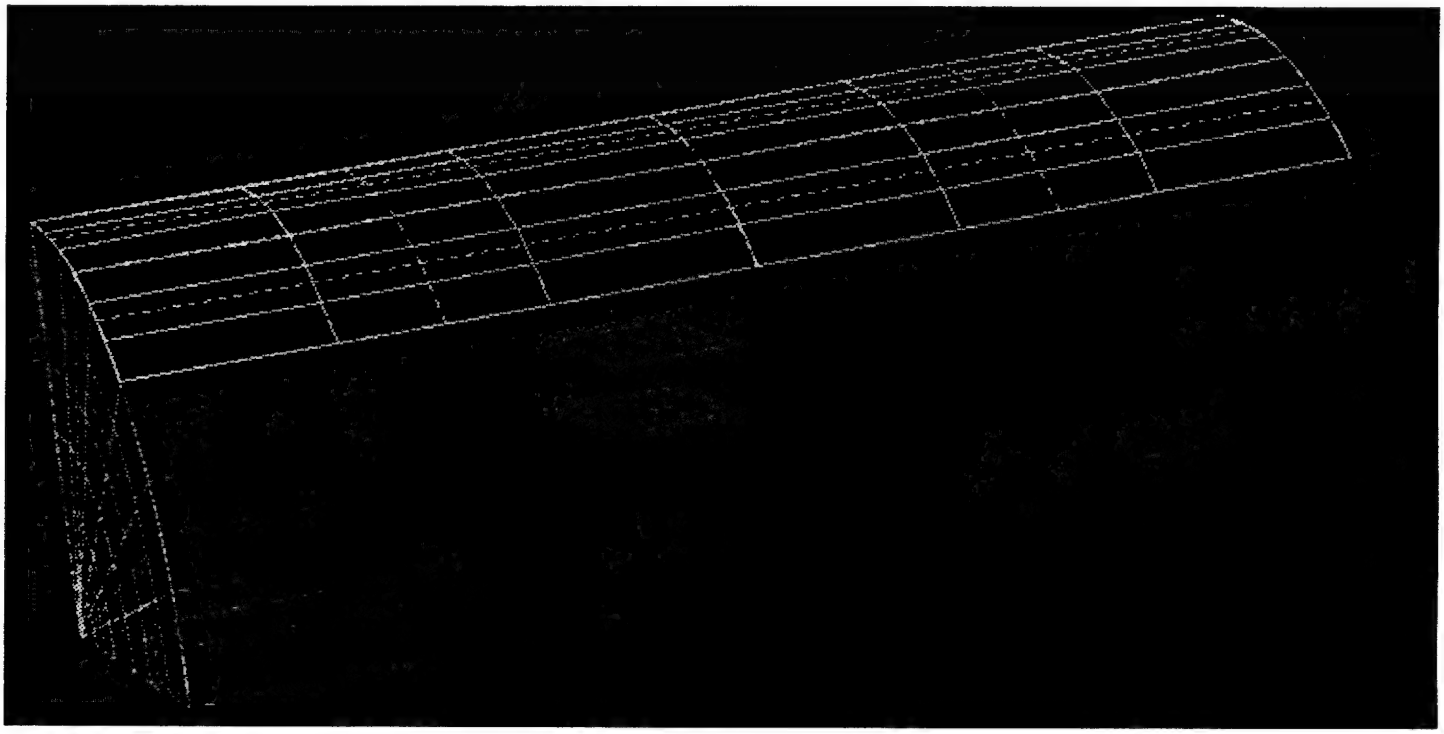




Step 3: Using the mouse, click the left button on the **GLOBAL** line.

Step 4: Now click the **RECALL** line.

Step 5: Enter the **internal.rcl** file name. This file contains the commands necessary to create the following model.



This model is a truncated cylinder (bottom green, side red, and orange top) with a flat nose (blue) and a box (purple) representing an internal component.

The following pages contain the RECALL file commands with editorial comments on the right side of each page.

internal.rcl

```
GLOBAL-COMMANDS
VIEW-OPTIONS
ROTATE
-90.000    180.00
ROTATE
10.000    20.000    10.000
CDRAW
QUIT
QUIT
DEFINE-GEOMETRY
POINT
ENTER
1
0.00000E+00
-10.000
#
QUIT
LINE
ENTER
1
1 2
#
TRANSLATE
2
1
0.00000E+00 3.0000
CDRAW
QUIT
SURFACE
PARAMETRIC
1
1 2
SWEEP
2
2
1
-45.000
DISPLAY-OPTIONS
LABEL
LINE
ON
CDRAW
SWEEP
3
5
1
-45.000
```

comments

* Global view control, rotate to make x–axis point to the left, the y–axis toward you and the z–axis up. Then a small rotation for aspect.

* Define Geometry, points are entered first. Point 1 at 0, 0, 0 point 2 at –10, 0, 0.

* Go to the line menu, make one straight line between points 1 and 2.

* Create line number 2. Translate line 1 by 3 units in the y–direction.

* Go to the surface menu. Create surface 1 (the bottom) with a parametric fit between line 1 and line 2.

* Create surface 2 (the side) by sweeping line 2 about line 1 –45 degrees.

* Turn on line labels to see which line to sweep to create the top surface. The CDRAW redraws the model with line labels.

* Sweep line 5 about line 1 by –45 degrees to create surface 3.

Note: Each surface is created to have normals pointing in the outward direction, generally.

internal.rcl continued

```
SWEEP
4
3
1
-45.000
DELETE
4
NO
#
SWEEP
4
3
1
-90.000
QUIT
QUIT
DEFINE-GEOMETRY
POINT
ENTER
19
-4.0000  0.00000E+00 1.5000
-4.0000    1.5000    1.5000
-5.0000    1.5000    1.5000
-5.0000  0.00000E+00 1.5000
-4.0000  0.00000E+00 1.0000
-4.0000    1.5000    1.0000
-5.0000    1.5000    1.0000
-5.0000  0.00000E+00 1.0000
#
QUIT
SURFACE
ENTER
5
19202122
23242526
#
QUIT
SOLID
PARAMETRIC
1
56
QUIT
QUIT
```

comments

* Create surface 4 (the nose) by sweeping line 3 about line 1 by −90 degrees. Note the error and deletion of the first attempt by −45 degrees.

* Return to the point menu. Input 8 points for the internal component. We choose to represent it with two surfaces that bound the needed volume.

* Enter two surface with four corner points apiece.

* Go to the solid menu. Create a solid with a parametric fit between the two surfaces just created. This solid is created just for visualization.

internal.rcl continued

```
GENERATE-MESH
CREATE
ELEMENT
SURFACE
1 3
LINEAR
QUADRILATERAL
EDGE
6 6
1.0000
1
1
s1
s2
s3
SURFACE
4
LINEAR
QUADRILATERAL
EDGE
6 6
1.0000
1
1
s4
SURFACE
5 6
LINEAR
QUADRILATERAL
EDGE
1 1
1.0000
1
1
i1
i2
CDRAW
QUIT
QUIT
QUIT
GLOBAL-COMMANDS
```

comments

* Go to the main menu and then to the Generate Mesh menu.

* Create finite element meshes on each surface. First, surfaces 1 thru 3 are meshed with 6 by 6 linear quad elements.
6 element divisions along the parametric i–direction and 6 elements along the j–direction.
* Label these s1, s2 and s3.
Note: If you use the right–hand–rule and cross the i–direction into the j–direction, your thumb will point in the general outward direction for that surface.

* Mesh the nose surface with a 6 by 6 mesh also. Label it s4.

* Mesh the two surface, 5 and 6, for the internal component.
Only a 1 by 1 mesh is needed for each of these surfaces.
* Label these i1 and i2.

*Return to Main Menu then to the Global Menu.

internal.rcl continued

```
DISPLAY-OPTIONS
LABEL
ALL
OFF
QUIT
ENTITY
ALL
OFF
GROUP
ON
ELEMENT
ON
CDRAW
QUIT
QUIT
QUIT
GLOBAL-COMMANDS
GROUPS
NEW
ADD
SURFACE
4
#
QUIT
SAVE
nose
BLUE
NEW
ADD
SURFACE
1
#
QUIT
SAVE
bottom
GREEN
NEW
ADD
SURFACE
2
#
QUIT
SAVE
side
RED
```

comments

* Go to the Display Options menu.

* Turn off all labels.

* Turn off all entities.

* Turn back on just the element and group entities.

* Draw a centered view of the model so far.

* Enter the Global Group menu.

* Define each group:

*The nose is surface 4. This set of commands groups the surfaces chosen along with the elements associated with them. Color it blue.

*Now the bottom is grouped with surface 1. It is colored green.

* The side is grouped and colored red.

internal.rcl continued

```
NEW
ADD
SURFACE
3
#
QUIT
SAVE
top
ORANGE
CDRAW
NEW
ADD
SURFACE
5 6
#
QUIT
SAVE
intcomp
PURPLE
CDRAW
QUIT
QUIT
```

comments

* Continue grouping.

* Define the top group. Color it orange.

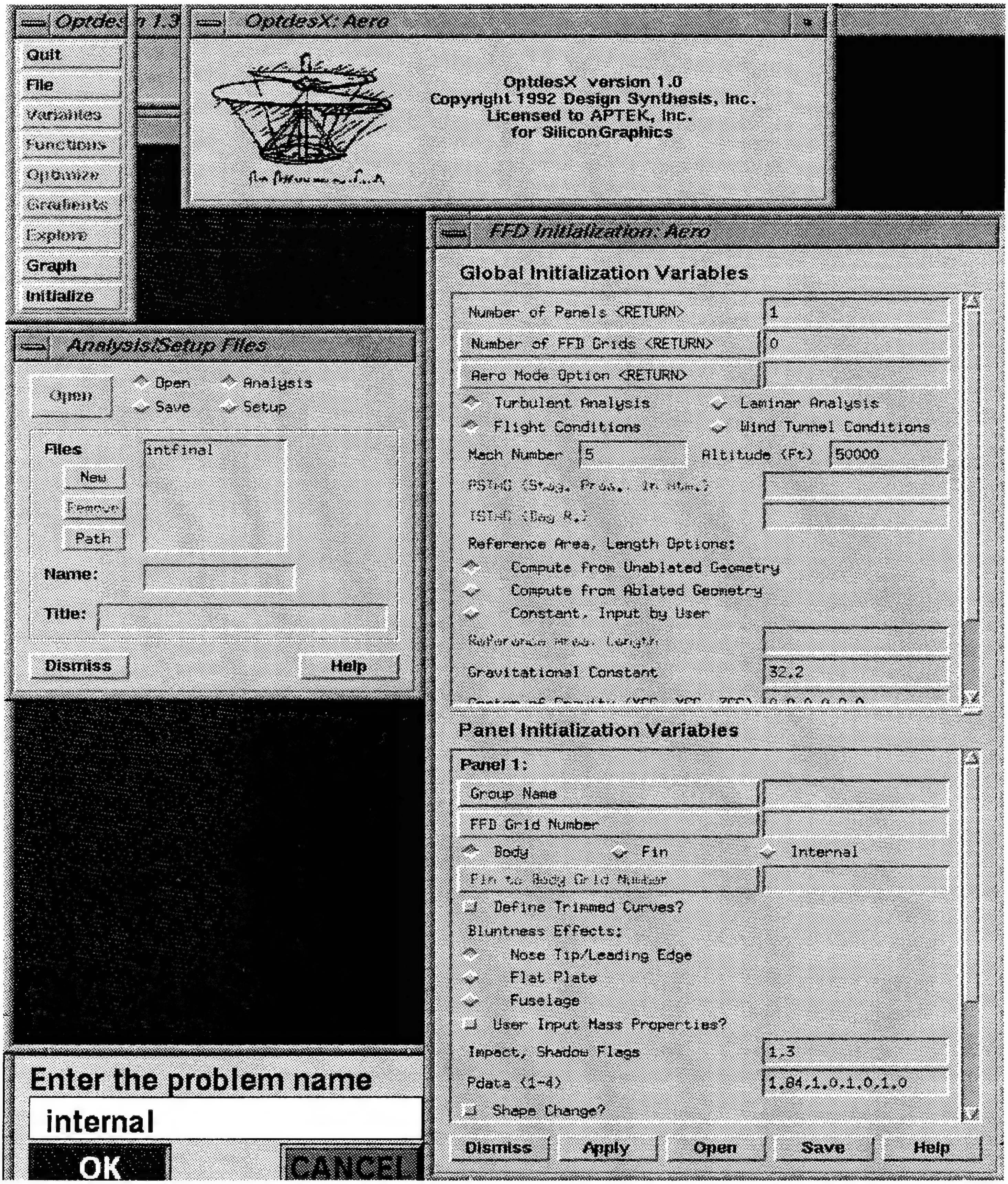
* Finally, group the internal component with surfaces 5 and 6. Color this one purple.

* Draw the model a last time to see if everything is okay.

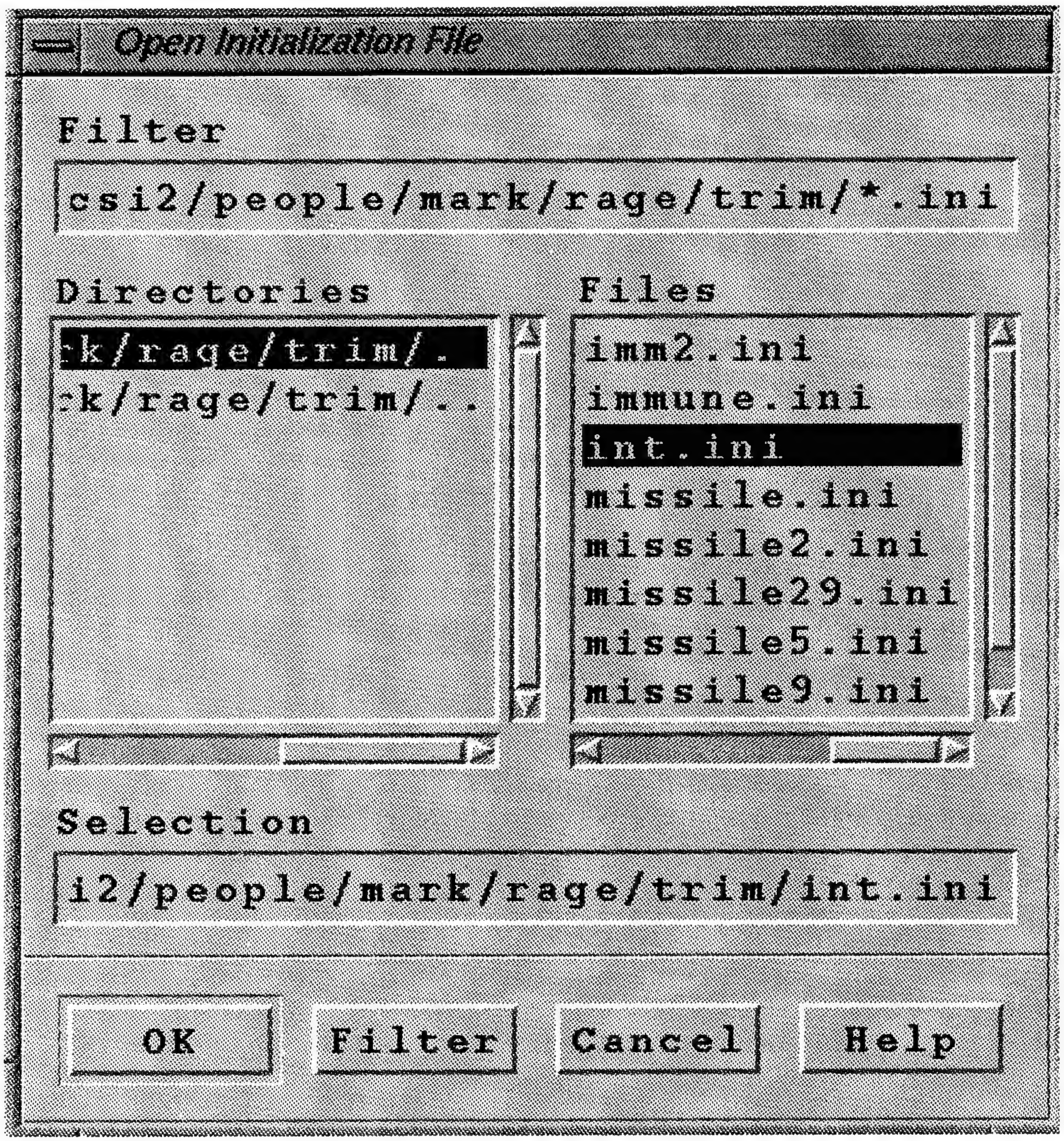
* This model is finished and is ready for **RAGE** and the shape optimization.

* Exit here and save the file as **internal**.

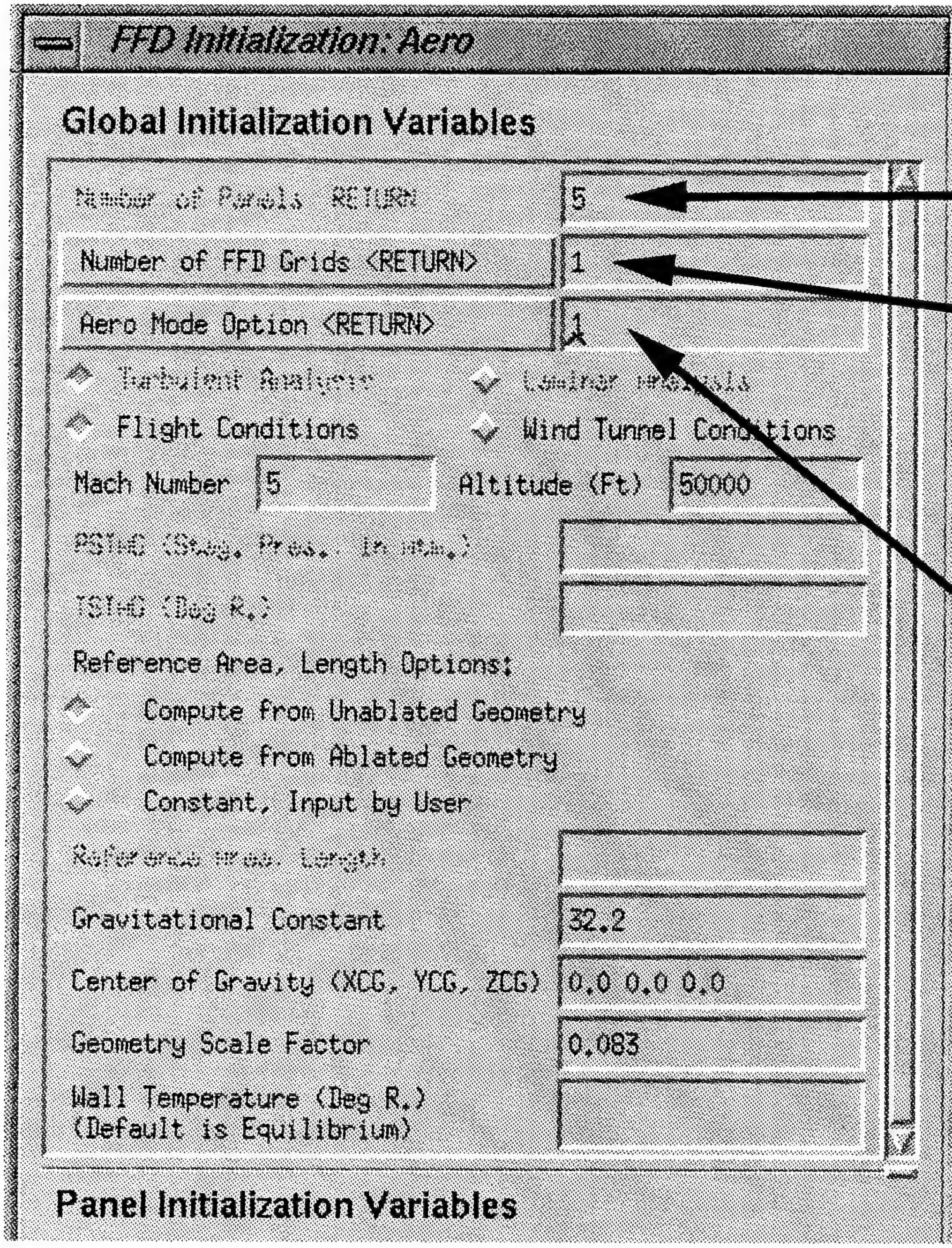
Step 6: Move mouse cursor into the **mvstart** menu. Type **r** for **RAGE**. The **Optdesx** window and the **Moviestar** file request will appear. Enter the **internal** file name here. Overwrite any .JOU and .LOG files. The **Optdesx** Setup Window and the Aero/FFD init. window appear.



Step 7: If you have saved an FFD/init file from a previous session, you may recall it by here double clicking the mouse on the file. Here it is **int.ini**. Otherwise, you may type in all of the necessary data as shown in the next window. The Open Init File is popped up by clicking on the OPEN button on the FFD/init file (at the bottom).



<u>Step 8:</u> Here we see the Global part of the FFD/init window. Note that there are many default values predefined. Change only those necessary to define your problem properly.



# of panels = 5.

# of FFD grids = 1. When you hit return in this window, a new window will popup to define the ffd grid.

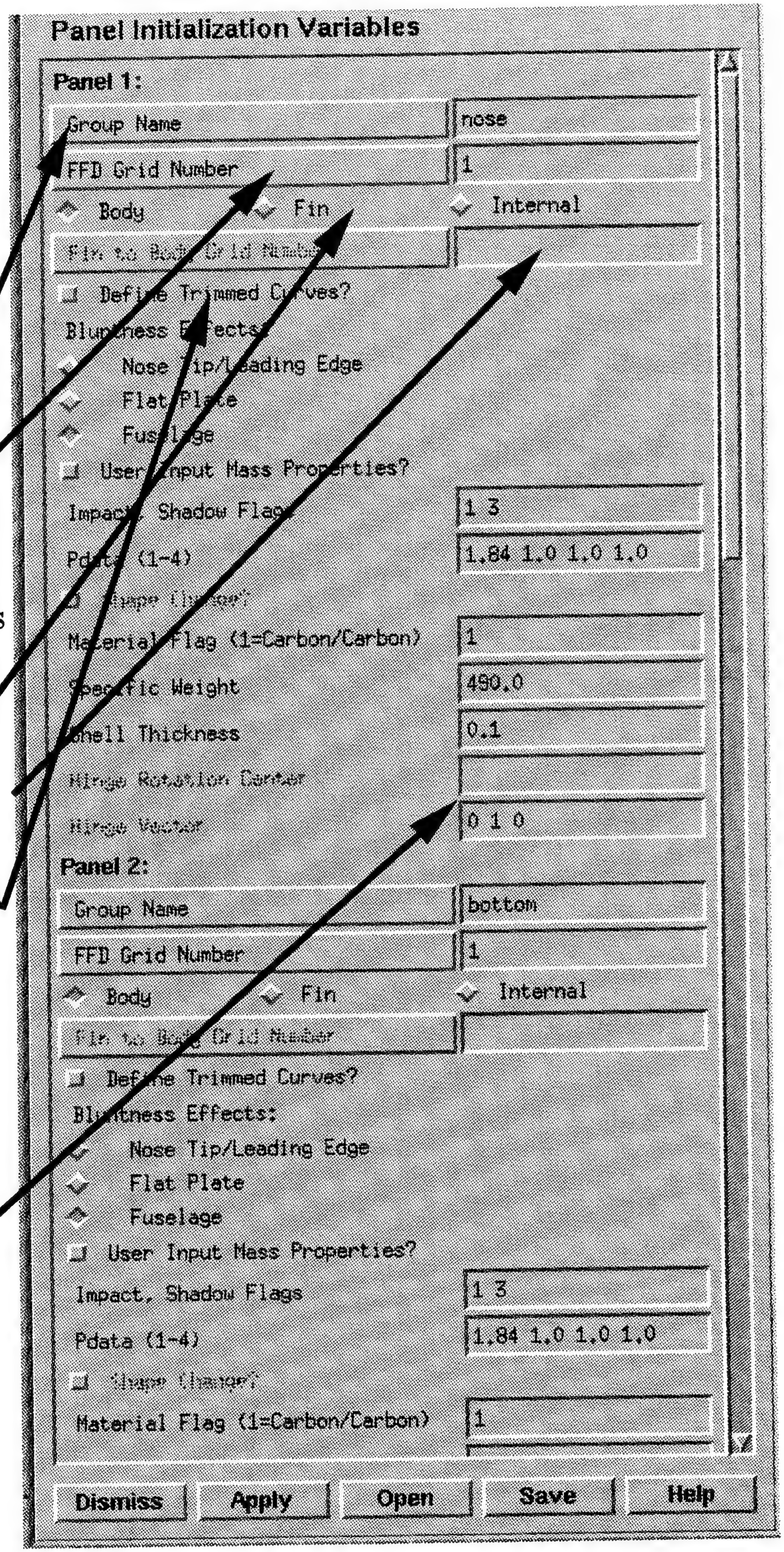
Aero mode option = 1 for inviscid aero calcs.

We accept all other default values.

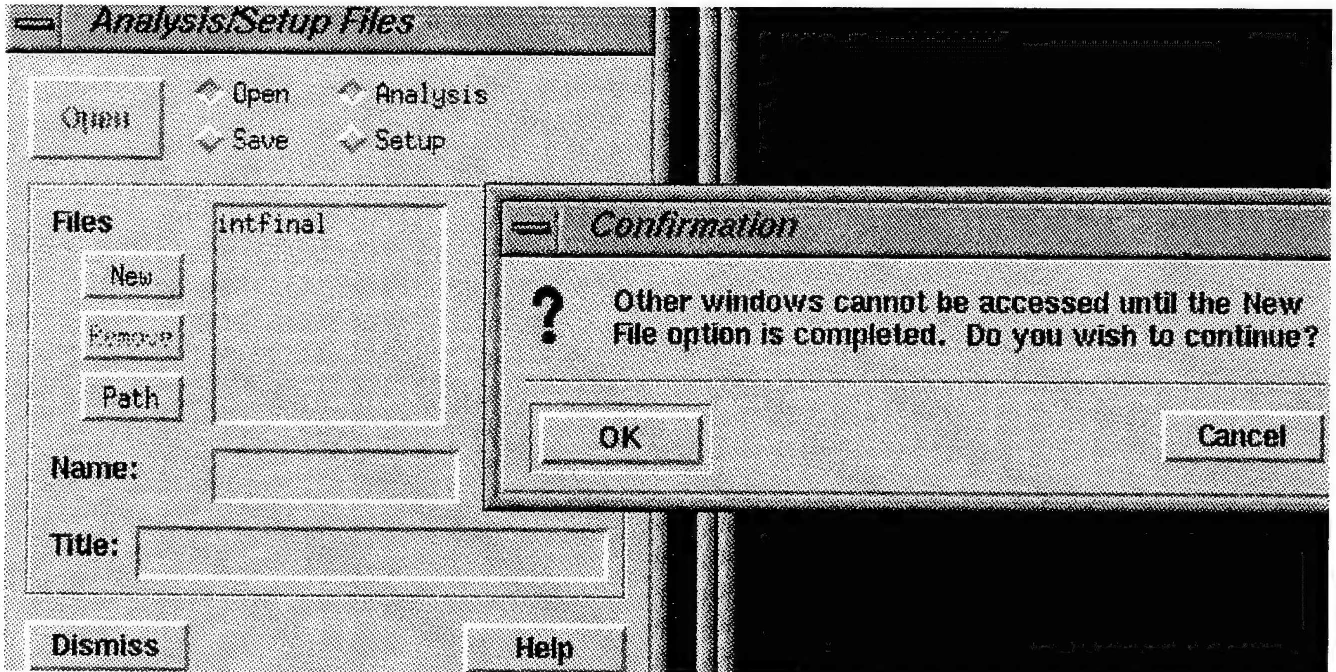
Step 9: The lower part of the FFD/init window needs data for each panel/group defined in **Moviestar**. We need to define each of the 5 panels.

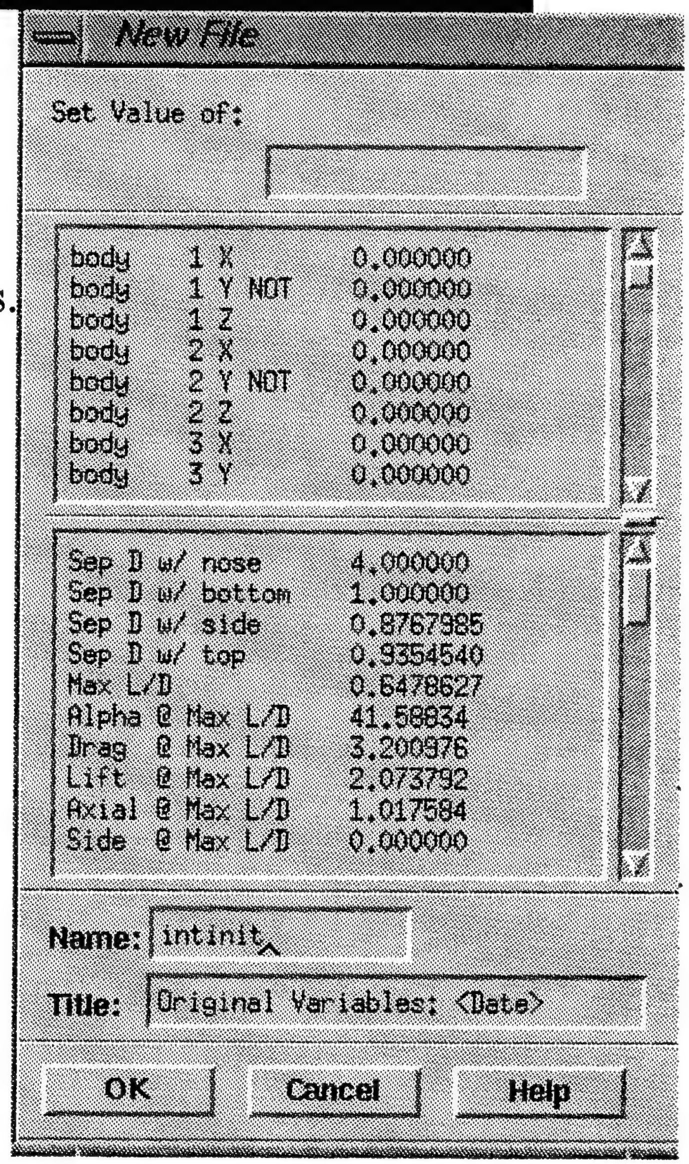
* Click on Group Name. A window of names will popup. Double click on the desired name.
* Click on FFD Grid #. A window of Grid names will popup. Double click on the desired Grid
* Click bod, fin, or int.
* If fin, enter fin to body FFD grid number.
* If trimmed aero–coeffs are desired, click here to popup window to define trimmed curves and immune node #s.
* All other data is used by HAB and/or Massproperties calcs.
* If fin, enter hinge center and vector.

Note: we have 4 body panels and 1 internal component panel.



Step 10: Click the **APPLY** button in the FFD/init window.
Step 11: Click the **NEW** button in the Analysis/Setup Files window. Then click **OK**. *IT IS BASICALLY ALL OPTDESX FROM HERE ON.*





Step 12: After the confirmation **OK**, the New File window will popup. Enter the initial values of all analysis variables. Usually all initial values are zero.
After a few second, the bottom portion of this window will fill with the values of all analysis functions.
If the values are satisfactory, click the **OK** button.
You may type a unique name and title before the **OK** click.
These files will be saved for later retrieval.

Analysis variables and analysis functions windows will now popup.

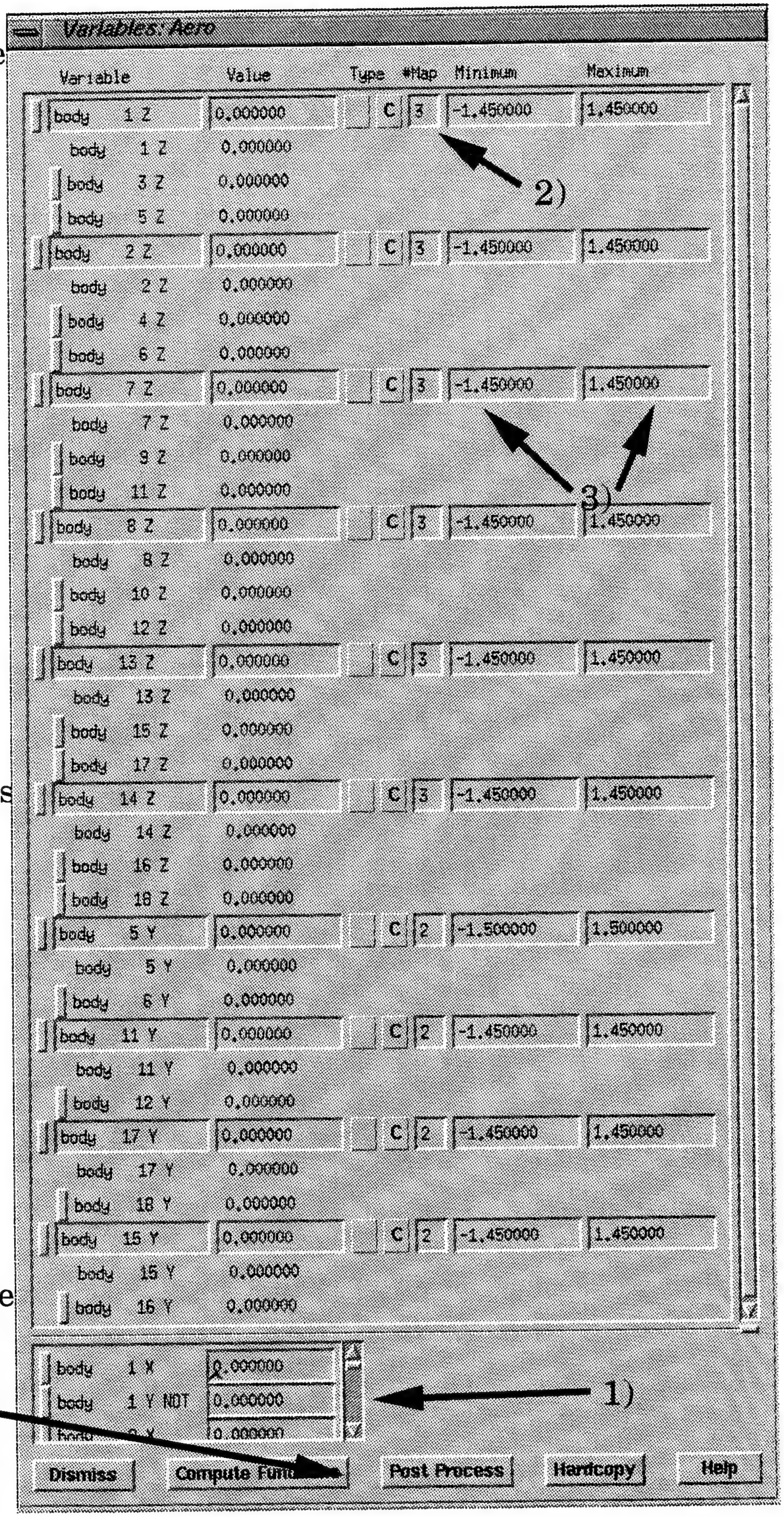
Step 13: Now we define the design problem by defining the desired design variables from the analysis variables.

1) We select from the lower window the design variables by clicking the left button associated with the analysis variable.

2) We may map several variables together by typing in the number of variables that are to be mapped here.

3) We can change the default values for the min and max limits for each design variable.

4) Note the buttons at the bottom for compute functions and post process.



Step 14: We now define the objective function and the constraints in the Function window.

1) Initially, all functions are in the lower window section.

2) Click on the left button of the function (**Max L/D here**) for the first design function. This will be the objective function.

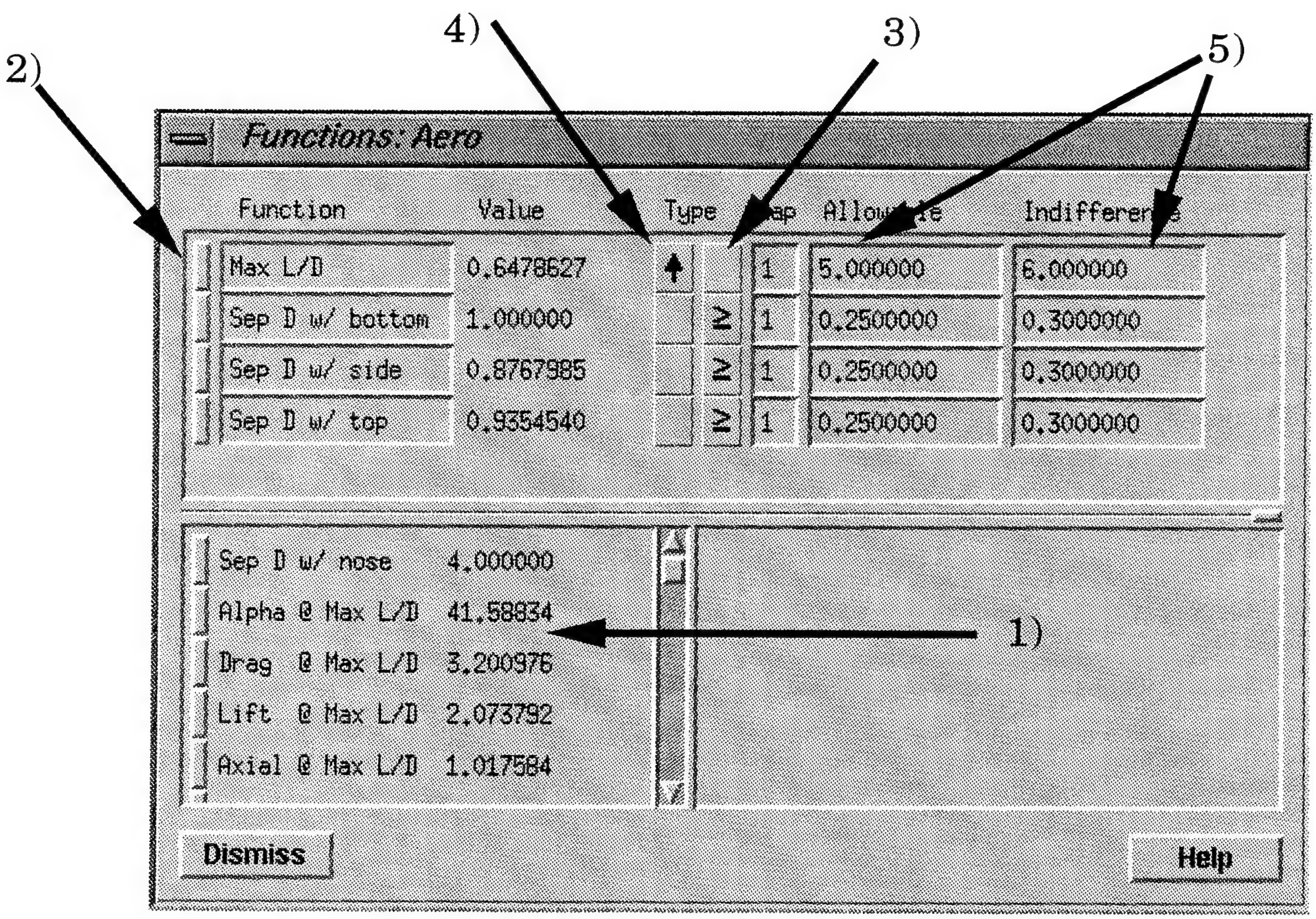
3) Click on the right Type button to remove the default constraint.

4) Click on the left Type botton to define as objective function.

5) Click on the separation distances w/ bottom, side and top.

6) Assign Allowable and Indifference values.

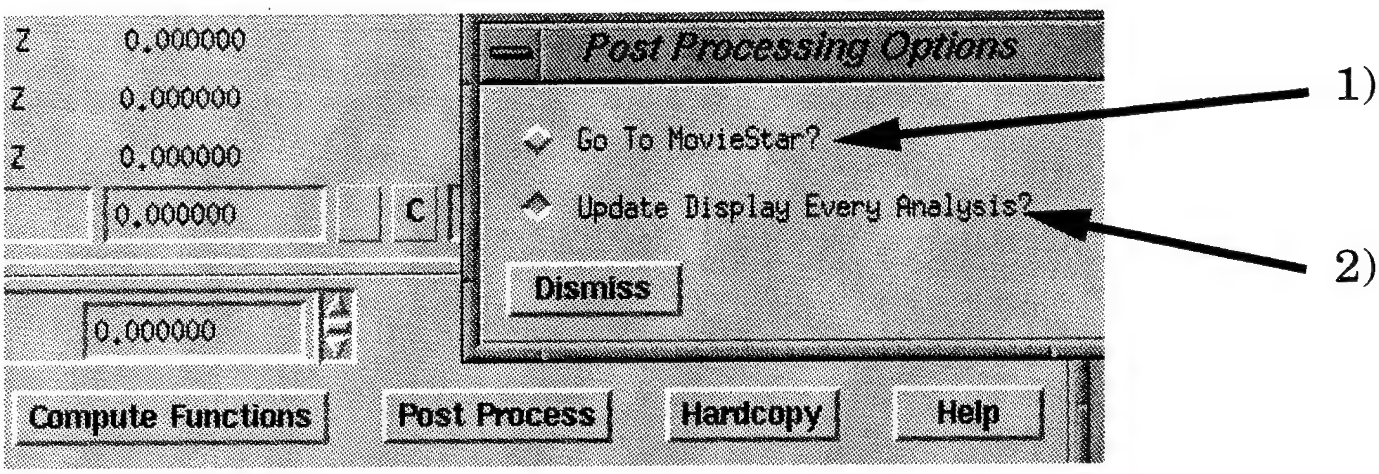
- For objective functions, we maximize if Indifference > Allowable and, we minimize if Allowable > Indifference.
- For constraints, we have **greater than** if Indifference > Allow. and we have **less than** if Allowable > Indifference.



Step 15: You may now optimize, explore design space, change variables one–by–one and compute functions, or post process. By clicking the post–process button a post process window pops up.

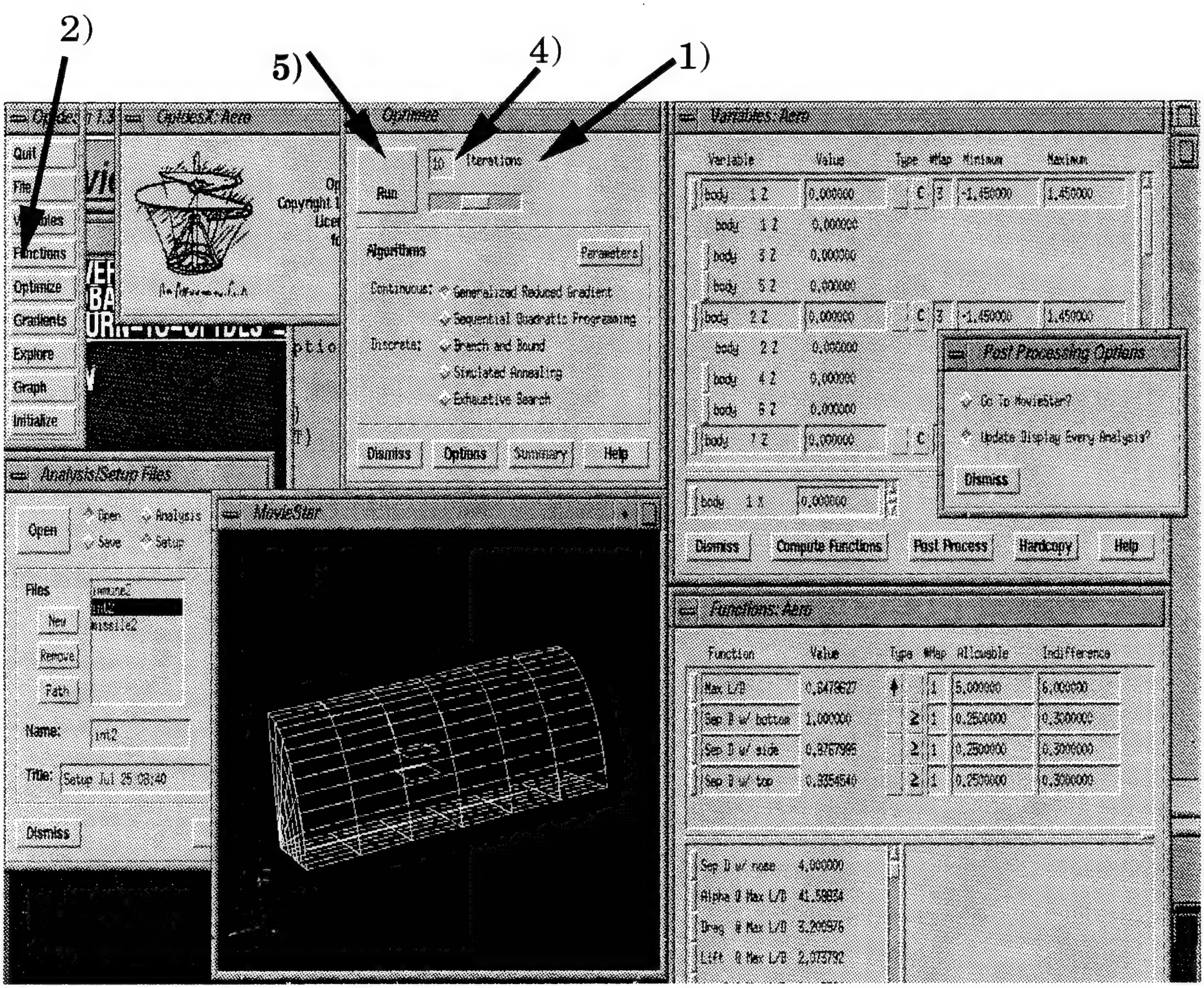
1) You can chose the **Go To Moviestar** to view with a **Moviestar** Global menu.

2) Or you can chose the **Update Display Every Analysis** button and view the design evolution at each analysis call. Originally the Moviestar graphics window is large, it is wise to click on its corner and resize it as shown on the next page.



Step 16: This page shows a complete screen after all is ready to go for optimization.

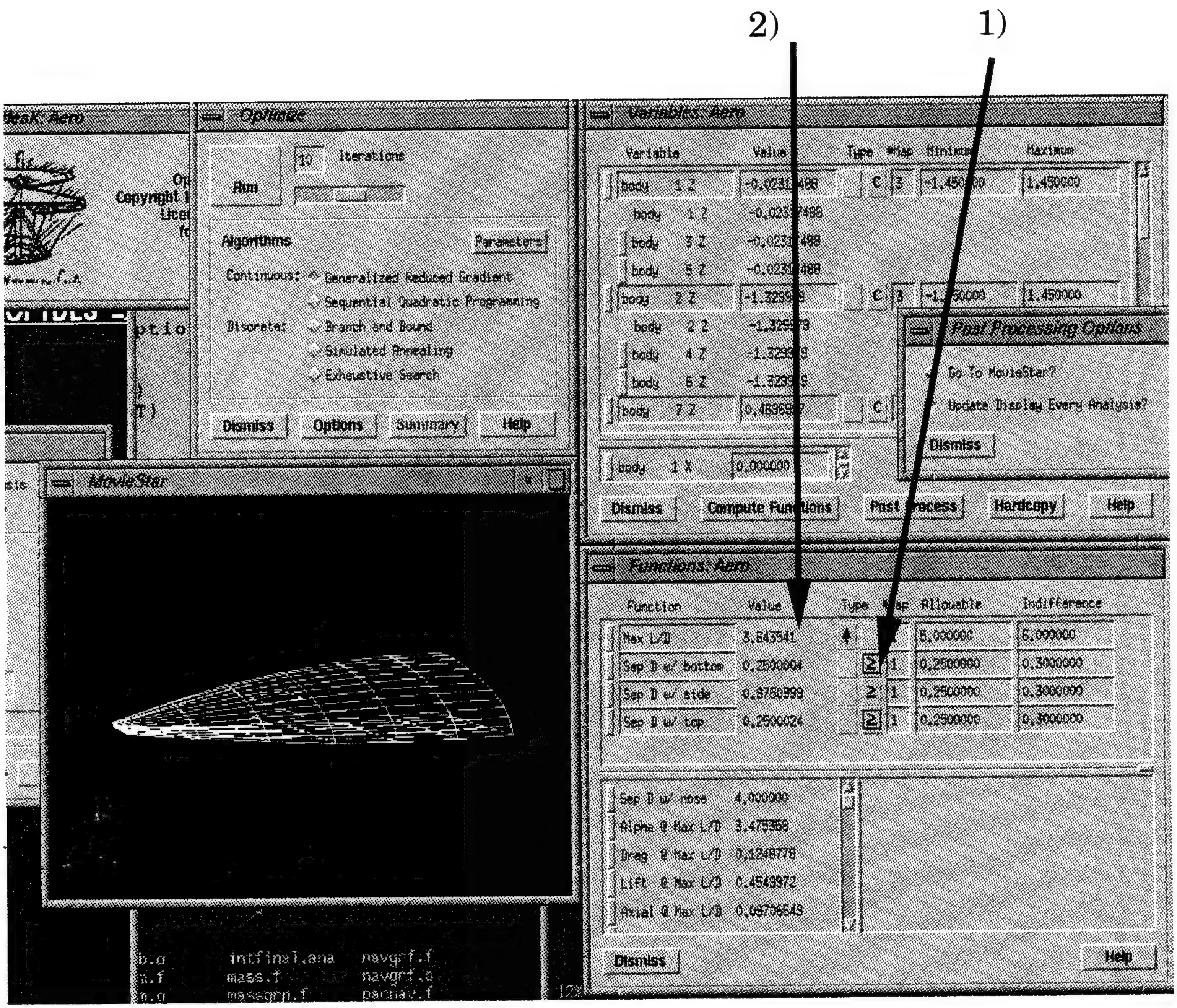
1) This is the Optimize window that pops up when you hit the Optimize button in the upper left corner of the screen.
2) Notice also the upper left corner has Graph, Explore, Etc...
3) Notice where the newly resized Moviestar graphics window is located.
4) Now you are ready to optimize by choosing the desired number of iterations.
5) Hit the **Run** button in the Optimize window.



Step 17: After about 4 iterations, we will see the following screen showing the final design variable and function values in the two right windows.

1) Note: the distances between the internal component and the top and bottom surfaces are at the constraint limits.

2) The objective function has improved from 0.65 to 3.64.



The second problem in this tutorial is to show the modeling and setup for a curve matching problem with trimmed aero–coeffs. The geometry is a simple missile with a cylindrical body, spherical nose and a fin.

The directions for this example will be less verbose than the first one.

The **RECALL** file (**immune.rcl**) on the following pages produces the model shown here.

Step 1: Make the model file in **Moviestar** as defined in the **RECALL** file **immune.rcl**.

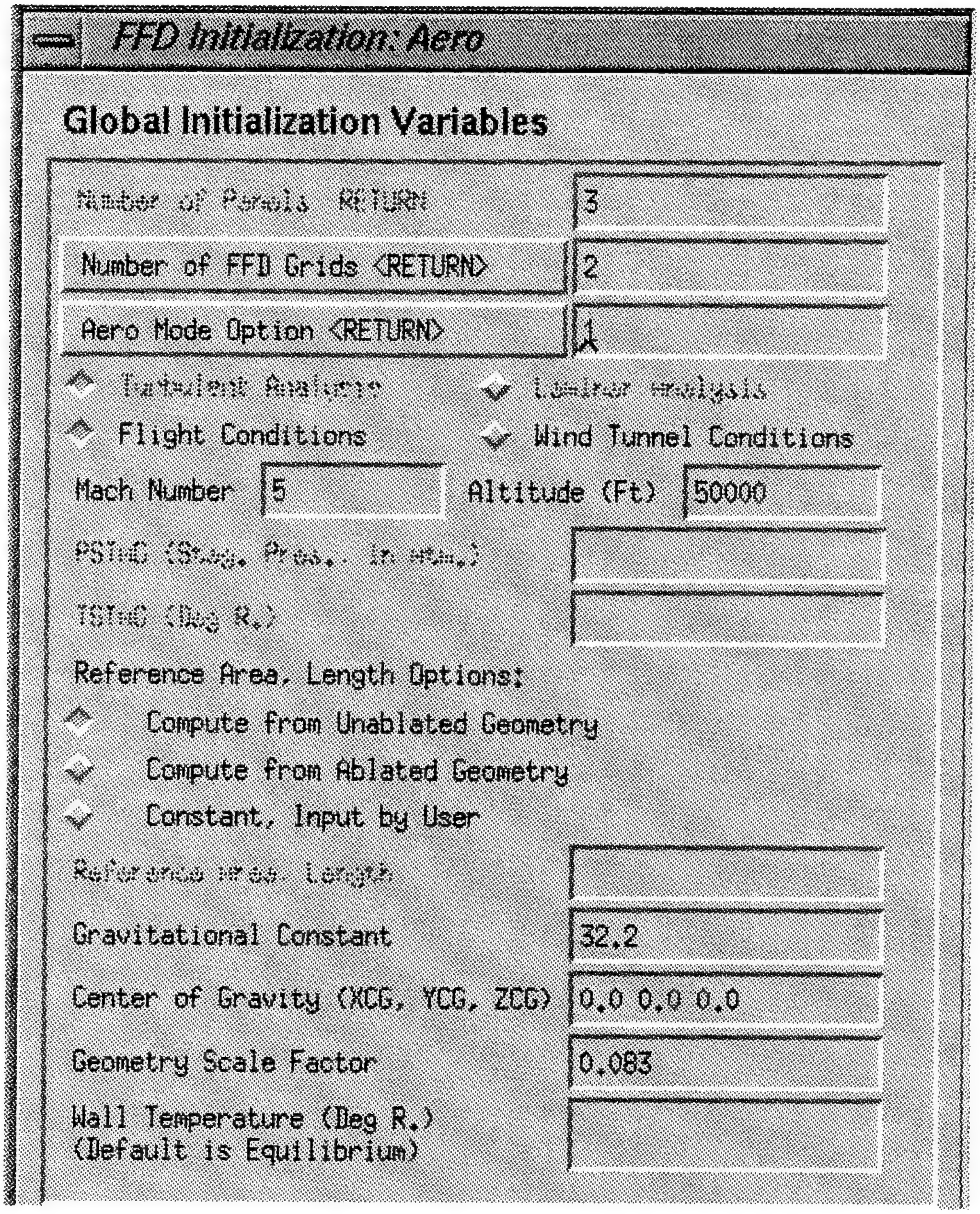
```
QUIT
GLOBAL-COMMANDS
VIEW-OPTIONS
ROTATE
-90.000        180.00
CDRAW
QUIT
QUIT
DEFINE-GEOMETRY
POINT
ENTER
1
0.00000E+00
-0.29289       0.00000E+00  -0.70711
-1.0000        0.00000E+00  -1.0000
-10.000        0.00000E+00  -1.0000
-10.000
-8.0000        1.0000
-8.0000        2.0000
-9.0000        1.0000        0.10000
-9.0000        2.0000        0.10000
-10.000        1.0000
-10.000        2.0000
-9.0000        1.0000       -0.10000
-9.0000        2.0000       -0.10000
#
CDRAW
QUIT
LINE
ENTER
1
1 5
#
ARC
2
1 2 3
ENTER
3
3 4
#
QUIT
SURFACE
SWEEP
1
2
1
-45.000
ROTATE
2
1
-135.00
3
DELETE
4
YES
YES
3
YES
YES
2
YES
YES
#
CDRAW
ROTATE
2
```

```
1
135.00
3
SWEEP
5
3
1
-45.000
ROTATE
6
5
135.00
3
ENTER
9
6 7 9 8
8 9 11 10
10 11 13 12
12 13 7 6
#
CDRAW
QUIT
QUIT
GENERATE-MESH
CREATE
ELEMENT
SURFACE
1 4
LINEAR
QUADRILATERAL
EDGE
6 3
1.0000
1
1
s1
s2
s3
s4
SURFACE
5 8
LINEAR
QUADRILATERAL
EDGE
20 3
1.0000
1
1
d1
d2
d3
d4
SURFACE
9 12
LINEAR
QUADRILATERAL
EDGE
1 1
1.0000
1
1
f1
f2
f3
f4
QUIT
```

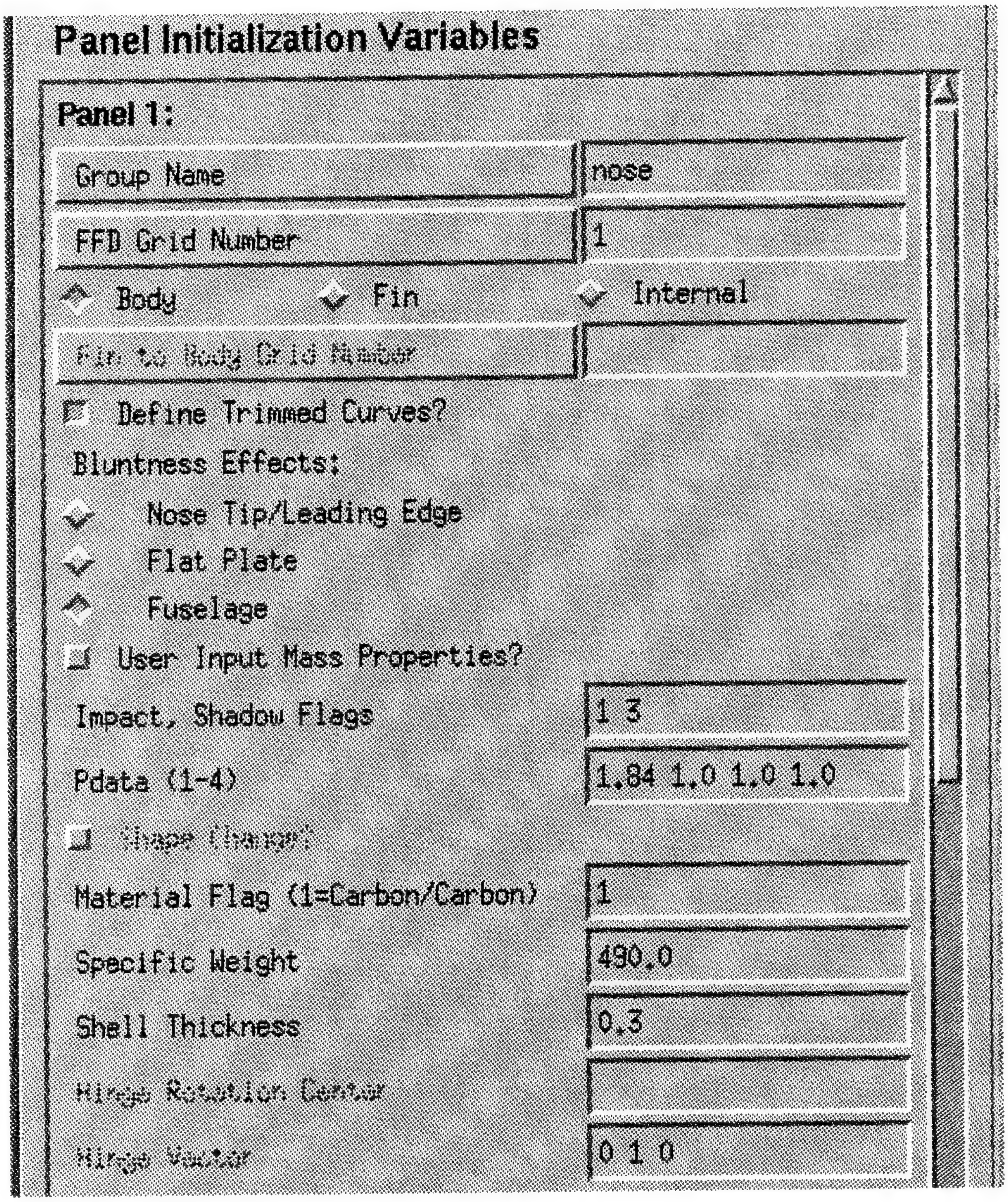
```
QUIT
QUIT
GLOBAL-COMMANDS
GROUPS
NEW
ADD
SURFACE
1 4
#
QUIT
SAVE
nose
RED
CDRAW
NEW
ADD
SURFACE
5 8
#
QUIT
SAVE
body
BLUE
CDRAW
NEW
ADD
SURFACE
9 12
#
QUIT
SAVE
fin
GREEN
QUIT
QUIT
GLOBAL-COMMANDS
DISPLAY-OPTIONS
ENTITY
ALL
OFF
GROUP
ON
CDRAW
QUIT
QUIT
QUIT
EXIT
```

Step 2: Run the **RAGE** executable by typing **r** in the **mvstart** menu.

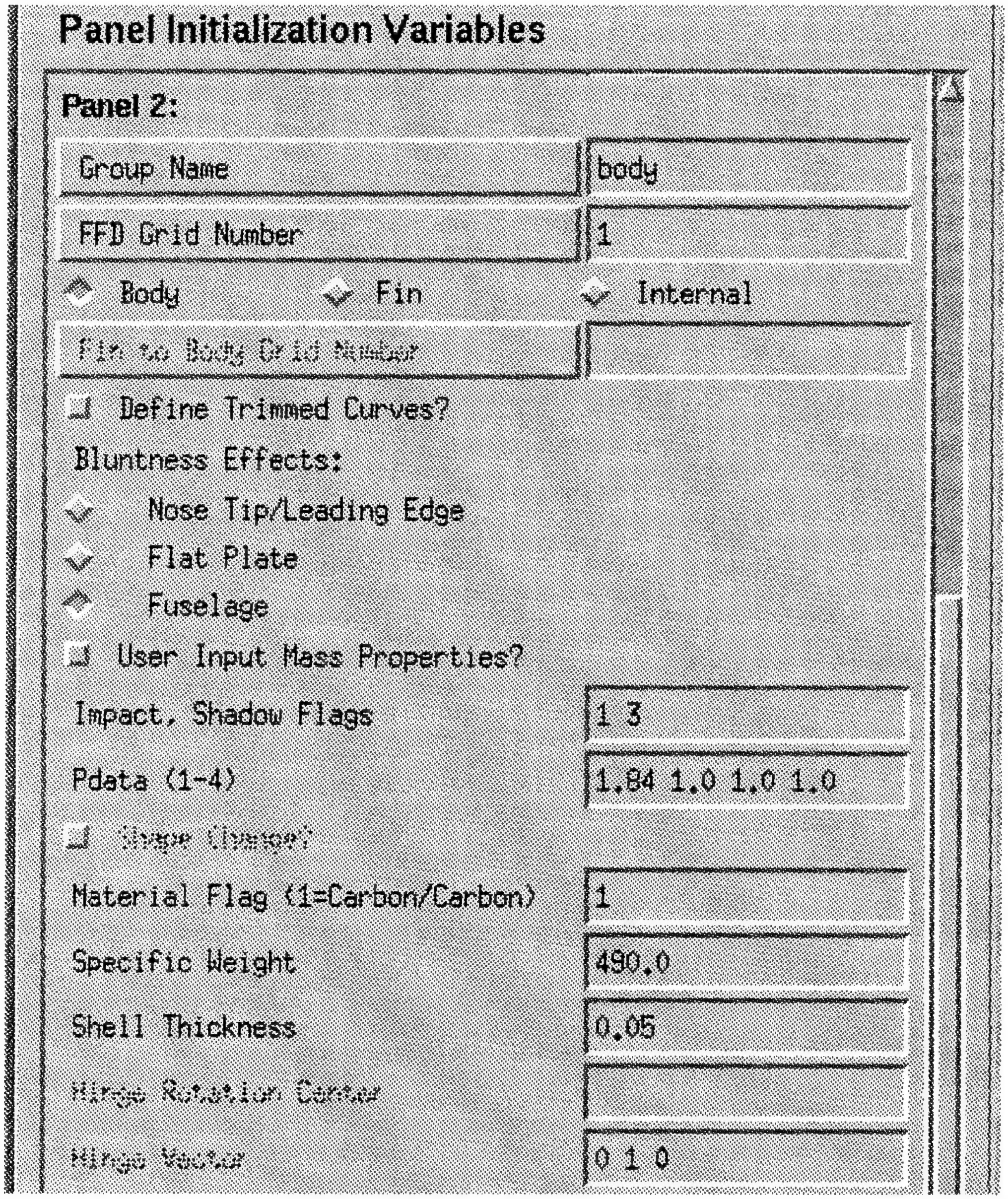
Step 3: The FFD/init Global Initialization should look like this before you continue. We have 3 panels, 2 FFD grids (one for the body and nose and the other for the fin).



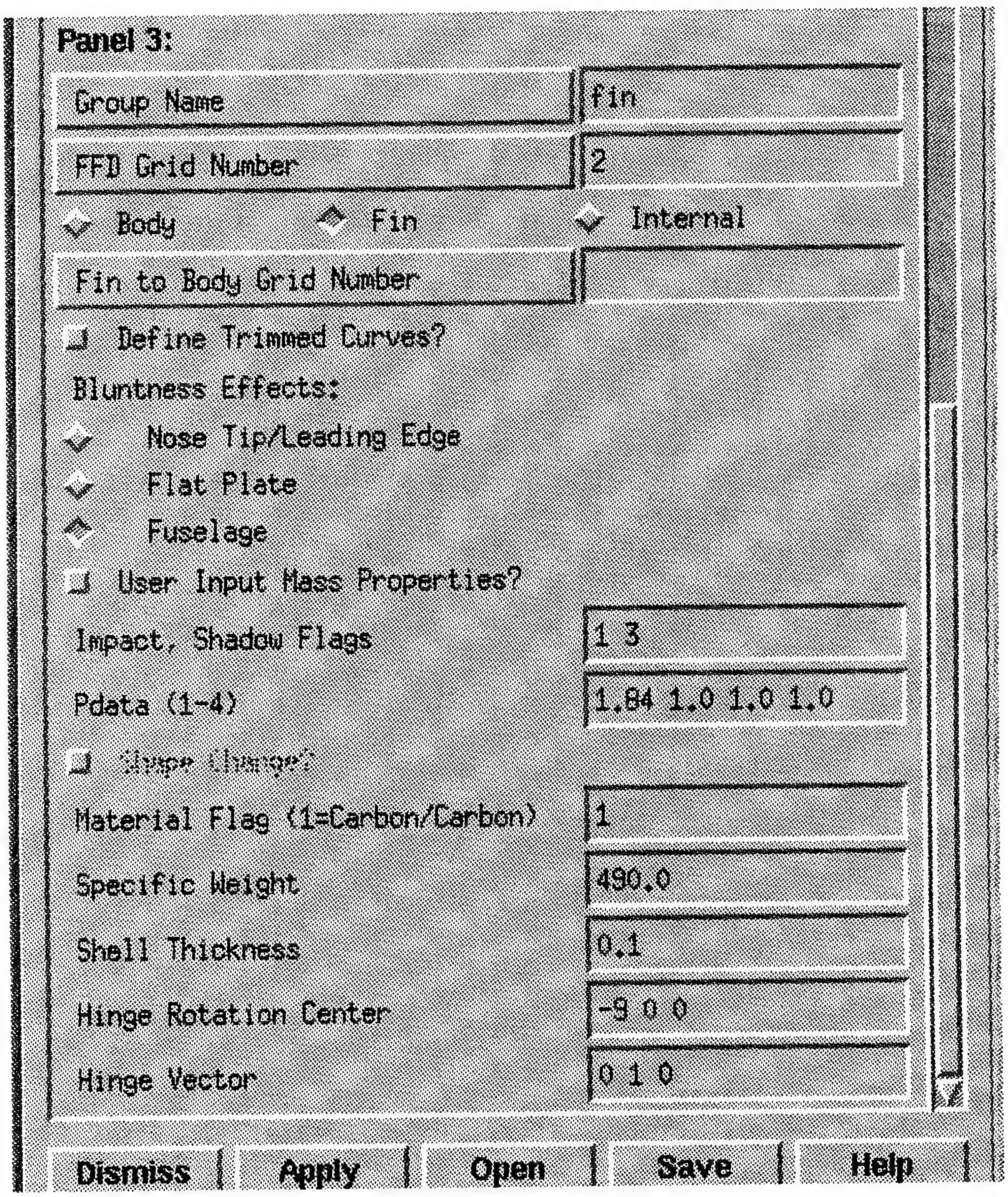
Step 4: The FFD/init panel 1 definition is the nose. The portion of the window should look like this.



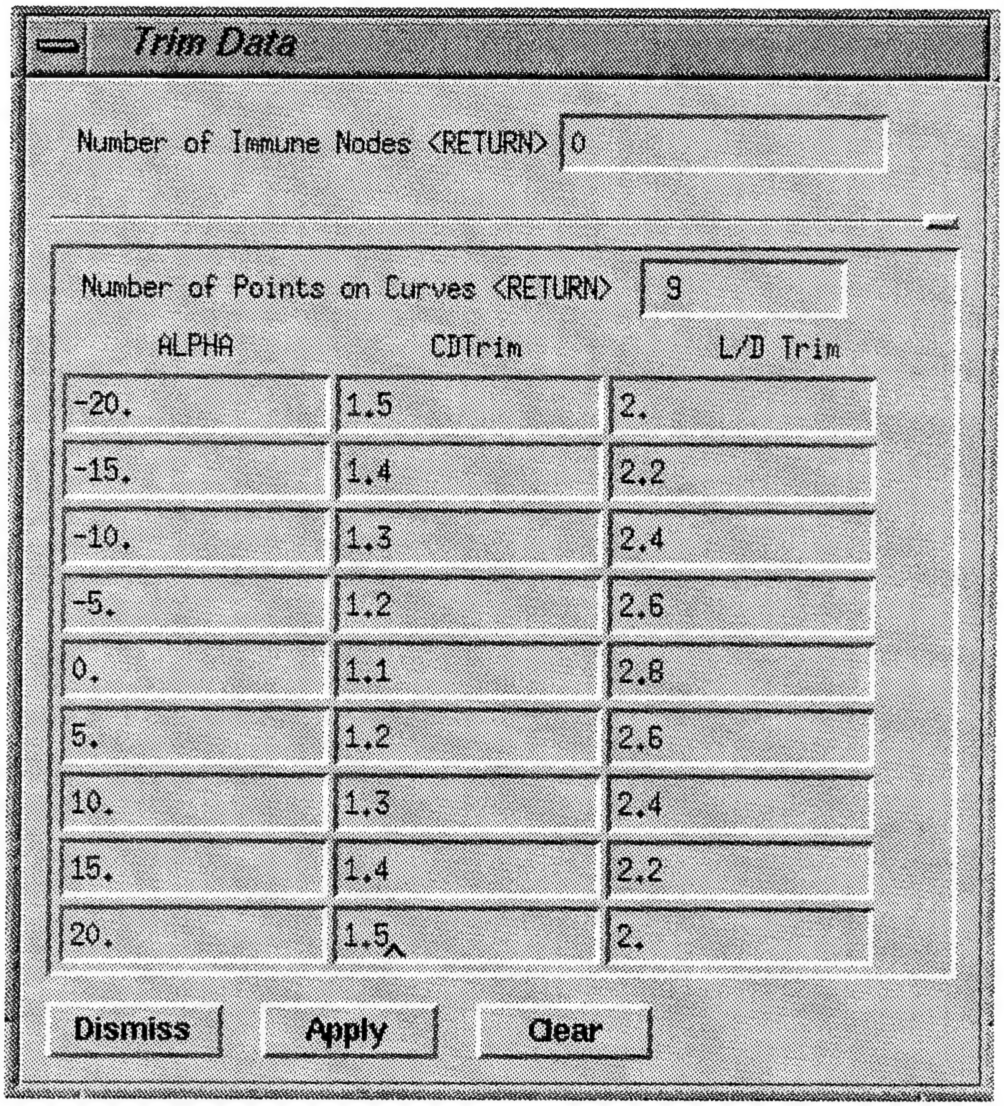
Step 5: Panel 2 (the body) should be defined and its window should look like this. Note that the nose and body share the same FFD grid.



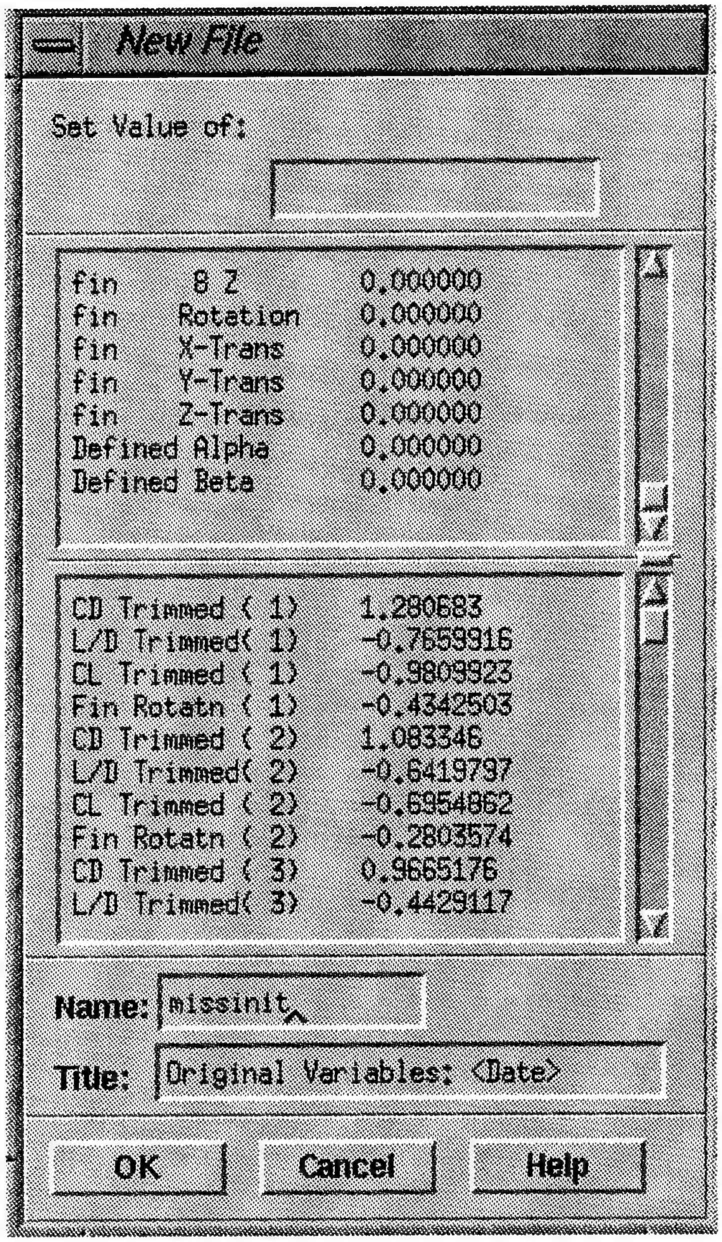
Step 6: The panel for the fin is defined as other panels but it also needs a hinge line. A hinge is defined with a center point and a vector about which the fin rotates. This fin's center is located at x = –9, therefore its center is (–9,0,0) and the vector is (0,1,0) pointing in the y–direction. The fin's panel window should look like this.



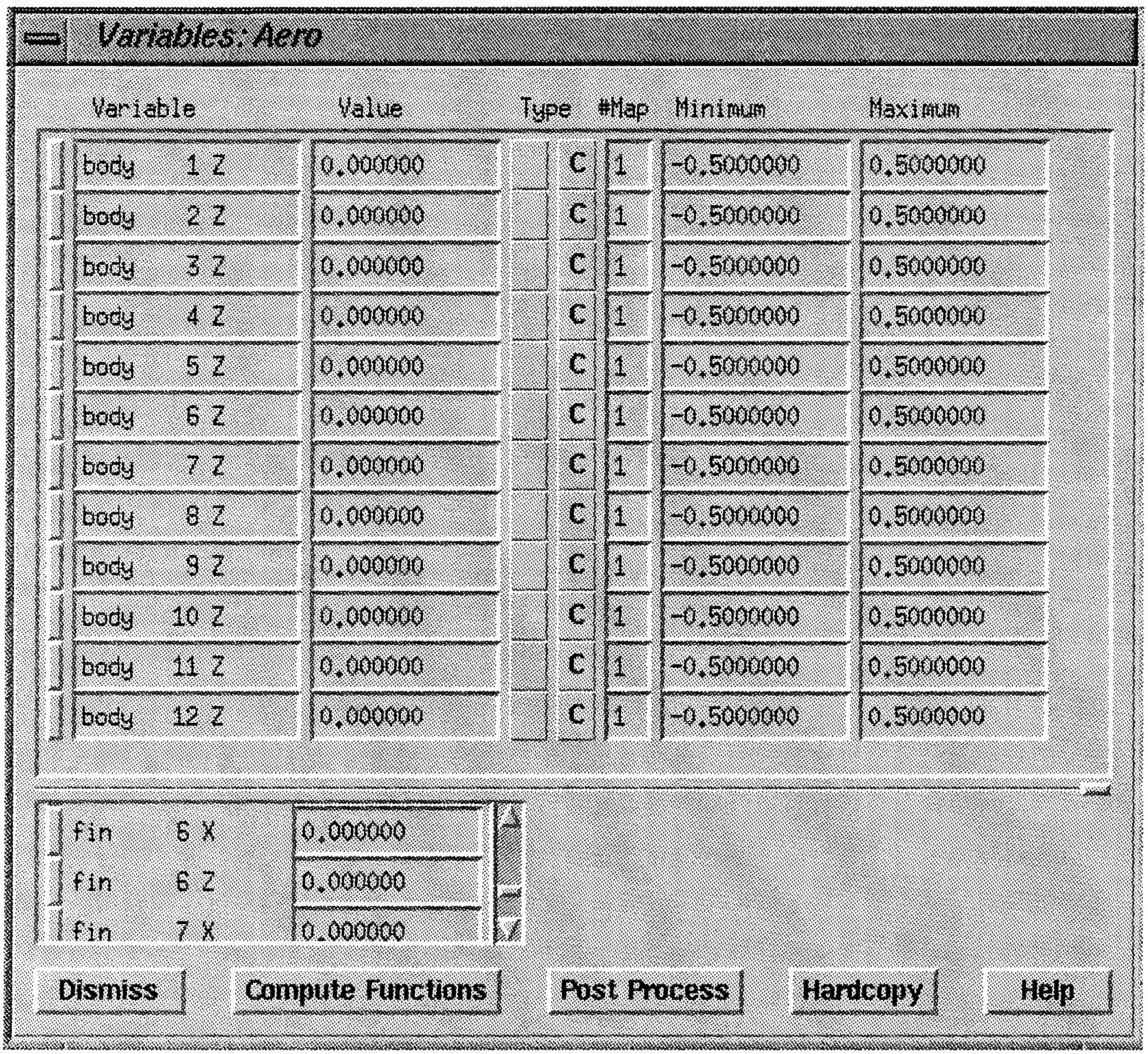
Step 7: The trim aero–curve data is defined after the **Defined Trimmed Curve?** button is pushed. Here you can optionally input the number of immune nodes (not for this example) and/or input the number of points on the trimmed curves. We choose 9 for this problem. After we enter 9, the window will change to allow entry of ALPHA, CDTrim, and L/D Trim for each point. The Trim Data window should look like this.



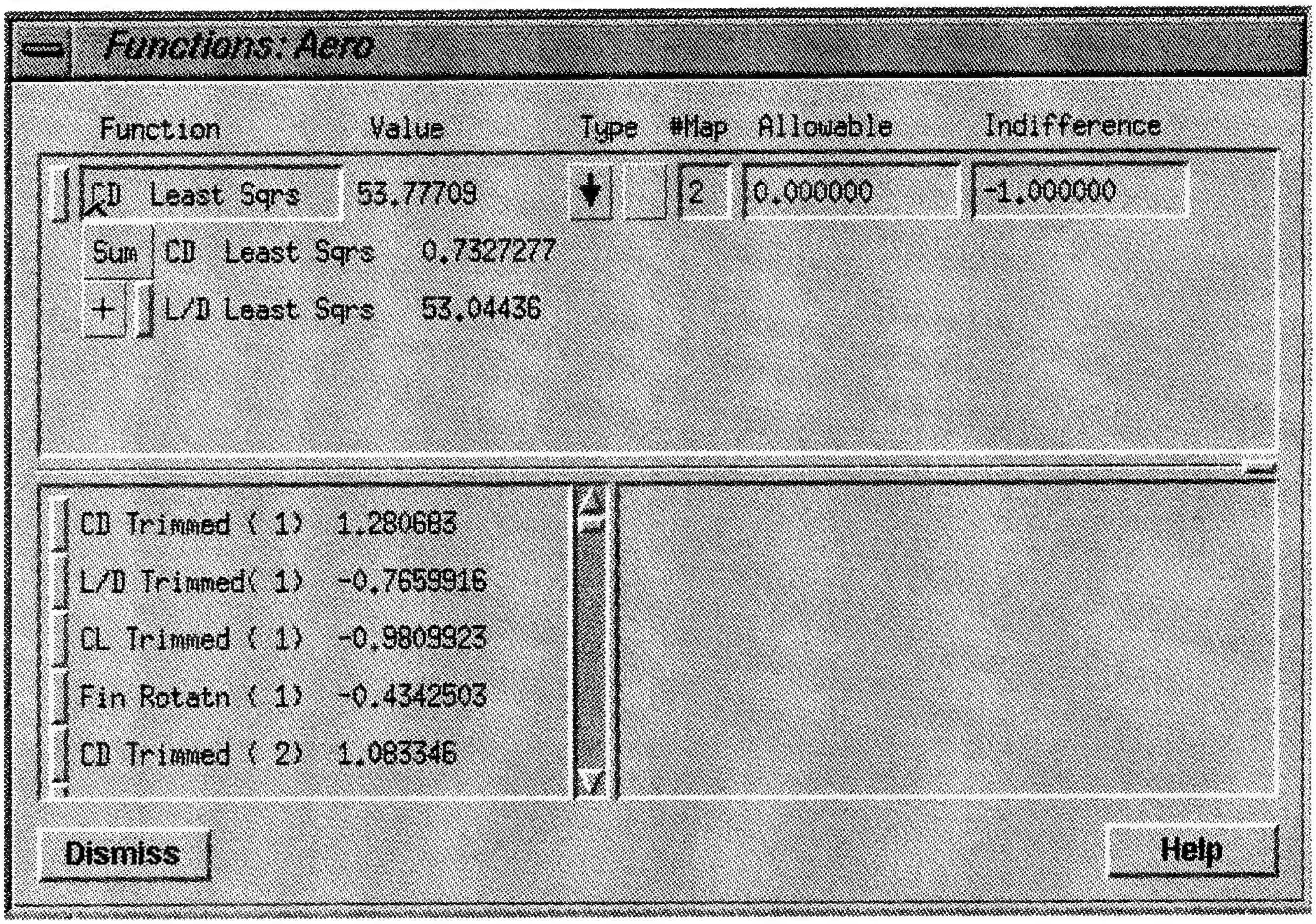
Step 8: The New File window is popped up to define the initial design.



Step 9: The Variable window and the Function window pops up. Define the design variables from the analysis variables as shown here. Note that each FFD coordinate chosen to vary is solitary (i.e. no mapping with other coordinates).



Step 10: The Function window is defined as shown here. Note that we mapped two functions into the objective function here. The sum option is used and we choose the Indifference value is less than the Allowable so that we minimize. We are attempting here to minimize the Least Square Difference between the current design's CD and L/D coefficients and the target coefficients as defined in the Trim Data window.

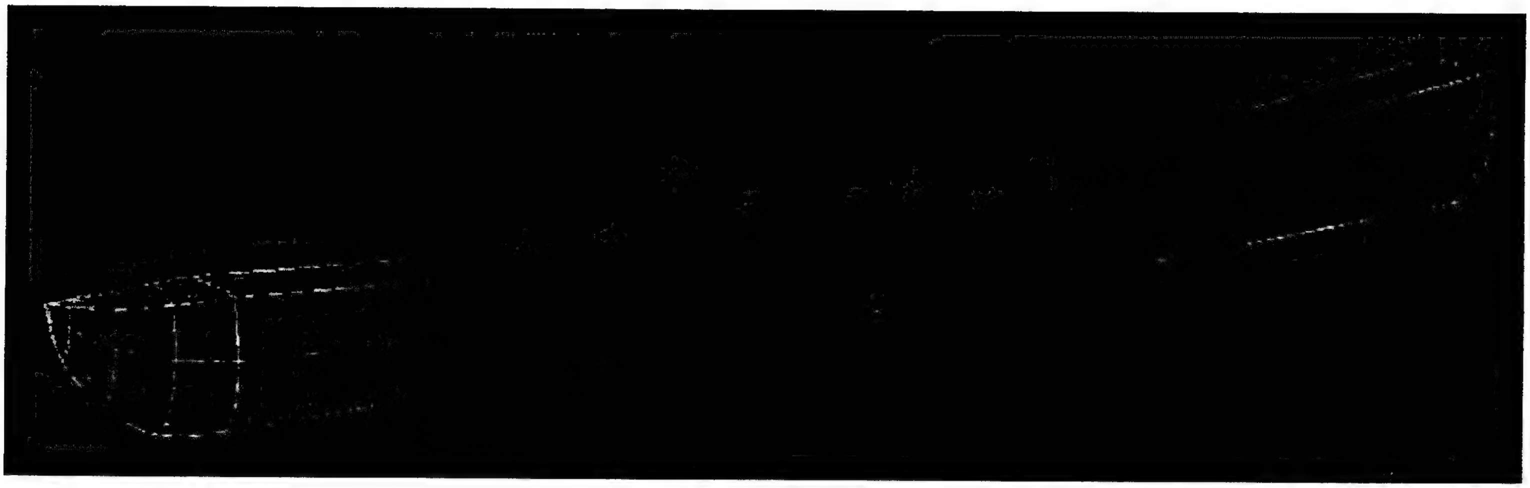


Step 11: Optimize by opening the Optimize window and choose the number of the iterations and then push the RUN button.

This third example shows how to model and setup a problem that will maximize the Lift to Drag ratio of a body while keeping certain user defined nodes immune to Free–Form Deformation. This will allow for shape optimization while retaining conformal surfaces.

Step 1: Execute the **mvstart** menu, type **c** to run **Moviestar**.

Step 2: The **RECALL** file found on the following pages will produce the following **Moviestar** model stored in file name **immune2.**



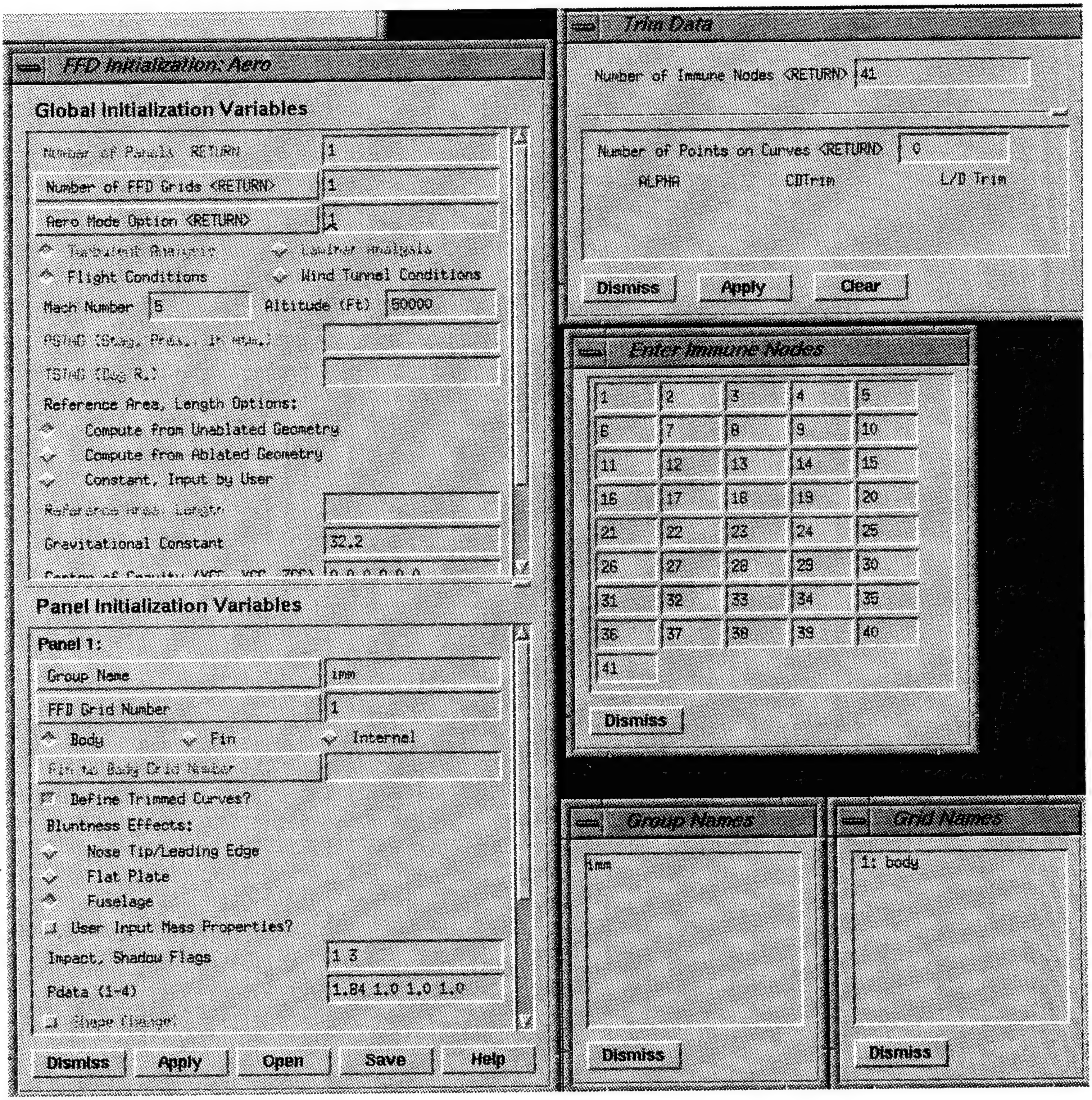
```
QUIT
VIEW-OPTIONS
ROTATE
-90.000         180.00
CDRAW
DEFINE-GEOMETRY
POINT
ENTER
1
0.00000E+00
-1.0000
-5.5000        0.00000E+00 -0.25000
-10.000
-0.29289        0.00000E+00 -0.70711
-1.0000        0.00000E+00  -1.0000
-10.000        0.00000E+00  -1.0000
#
CDRAW
QUIT
QUIT
DEFINE-GEOMETRY
LINE
ENTER
1
1 4
1 2
6 7
#
ARC
4
2 3 4
ARC
5
1 5 6
TRANSLATE
6
4
0.00000E+00   1.0000
CDRAW
VIEW-OPTIONS
ROTATE
20.000         30.000
CDRAW
ROTATE
7
3
45.000
1
ROTATE
8
5
45.000
1
ROTATE
9
8
45.000
1
QUIT
SURFACE
CDRAW
PARAMETRIC
1
#
DISPLAY-OPTIONS
LABEL
```

```
LINE
ON
CDRAW
PARAMETRIC
1
9 2
PARAMETRIC
2
6 4
PARAMETRIC
3
5 8
PARAMETRIC
4
8 9
CDRAW
DELETE
4
NO
3
NO
#
CDRAW
PARAMETRIC
3
5 8 9
PARAMETRIC
4
3 7 6
VIEW-OPTIONS
RESTORE
VIEW-OPTIONS
ROTATE
-90.000        180.00
CDRAW
VIEW-OPTIONS
ROTATE
0.00000E+00    90.000
CDRAW
QUIT
QUIT
GLOBAL-COMMANDS
DISPLAY-OPTIONS
ENTITY
ALL
OFF
ELEMENT
ON
GROUP
ON
SURFACE
ON
CDRAW
QUIT
QUIT
QUIT
VIEW-OPTIONS
RESTORE
VIEW-OPTIONS
ROTATE
-90.000        180.00
CDRAW
DEFINE-GEOMETRY
QUIT
GENERATE-MESH
CREATE
```

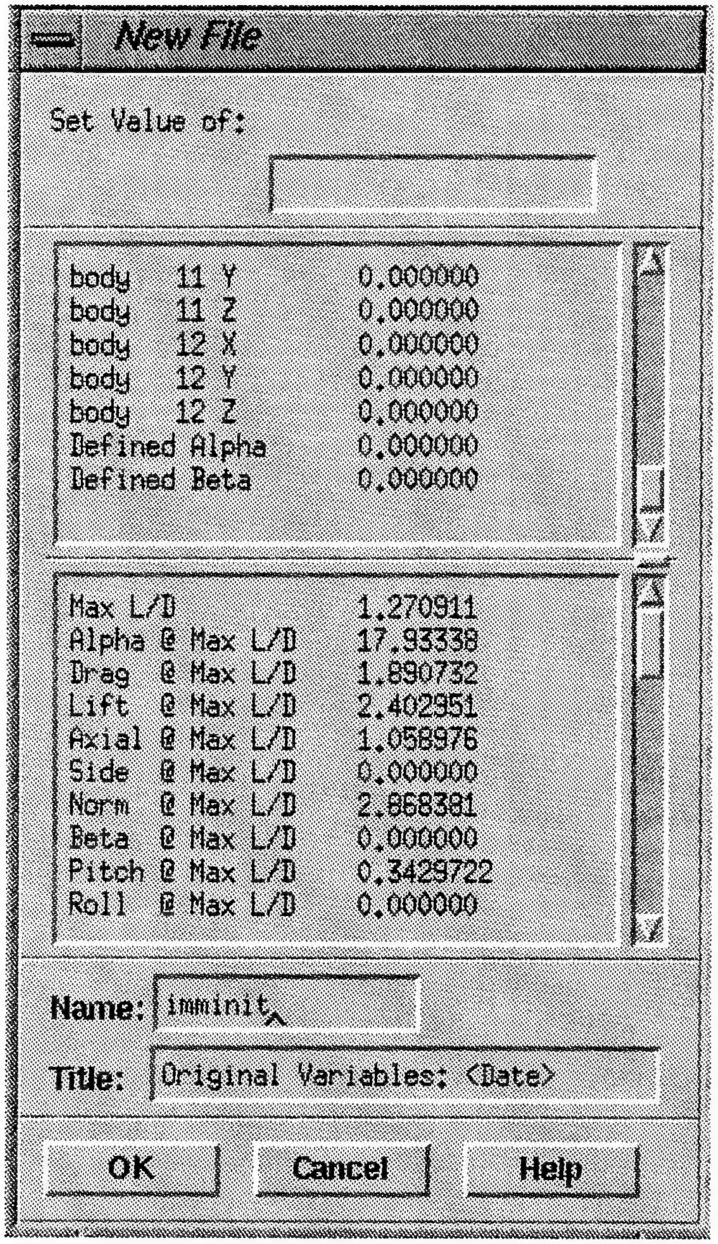
```
ELEMENT
SURFACE
1
LINEAR
QUADRILATERAL
EDGE
4 4
1.0000
1
1
s1
PLOT
QUIT
SURFACE
2
LINEAR
QUADRILATERAL
EDGE
4 4
1.0000
1
1
s2
SURFACE
3
LINEAR
QUADRILATERAL
EDGE
4 4
1.0000
1
1
s3
SURFACE
4
LINEAR
QUADRILATERAL
EDGE
4 4
1.0000
1
1
s4
QUIT
QUIT
QUIT
GLOBAL-COMMANDS
GROUPS
NEW
ADD
SURFACE
1 4
#
QUIT
SAVE
imm
RED
CDRAW
QUIT
VIEW-OPTIONS
ROTATE
10.000          30.000          10.000
CDRAW
QUIT
LIST
NODE
```

```
1 100



QUIT
DISPLAY-OPTIONS
ENTITY
NODE
ON
QUIT
LABEL
NODE
ON
CDRAW
NODE
OFF
CDRAW
QUIT
ENTITY
NODE
OFF
QUIT
QUIT
QUIT
CDRAW
```

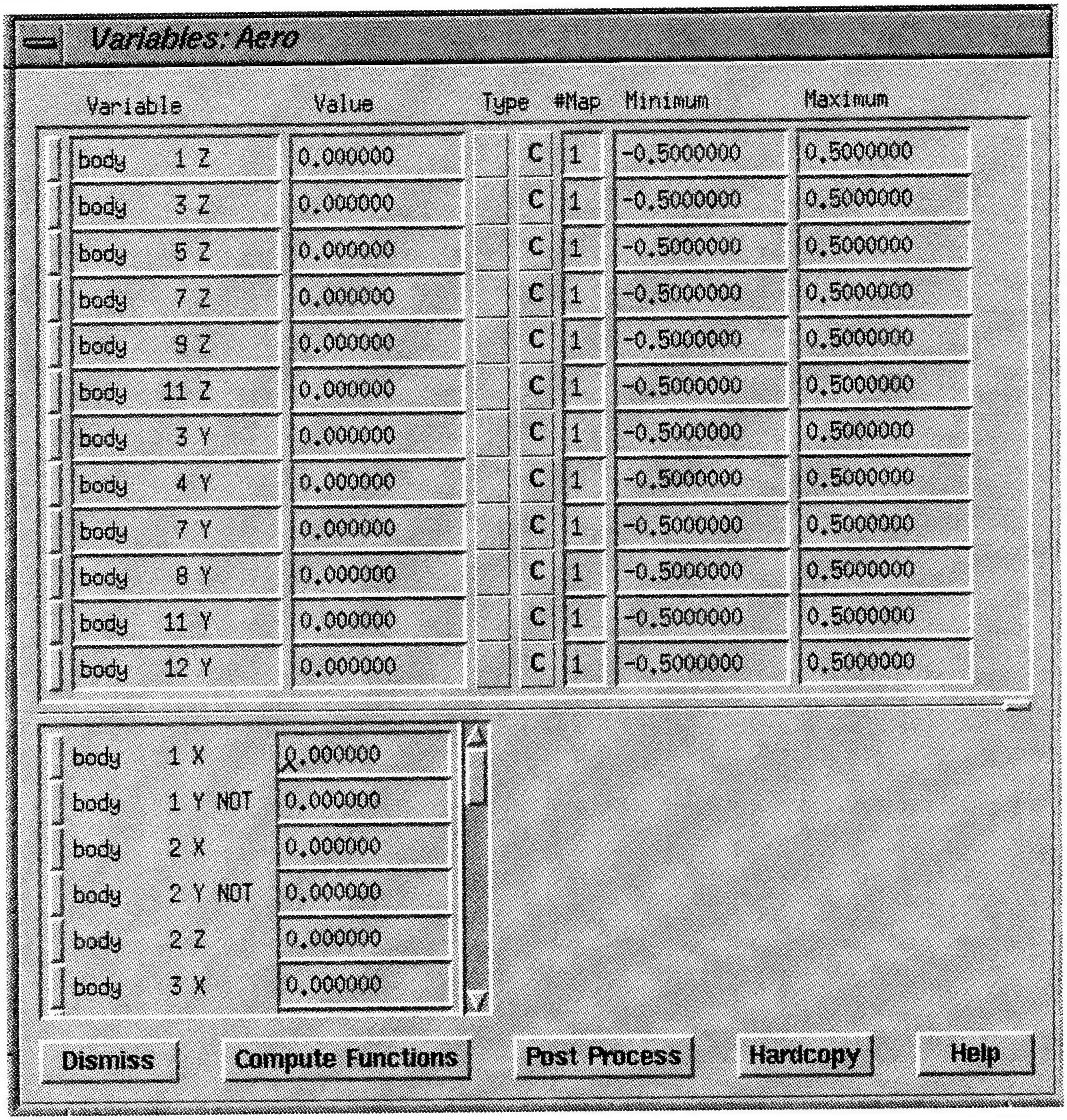
Step 3: The FFD/init and Trim Data window should be defined as follows. Note that there are no Trim Data points defined, but the Immune Node Numbers are defined. You get the node numbers in **Moviestar** by toggling the Node Labels ON. Also shown are the Group Name and FFD Grid Name windows.



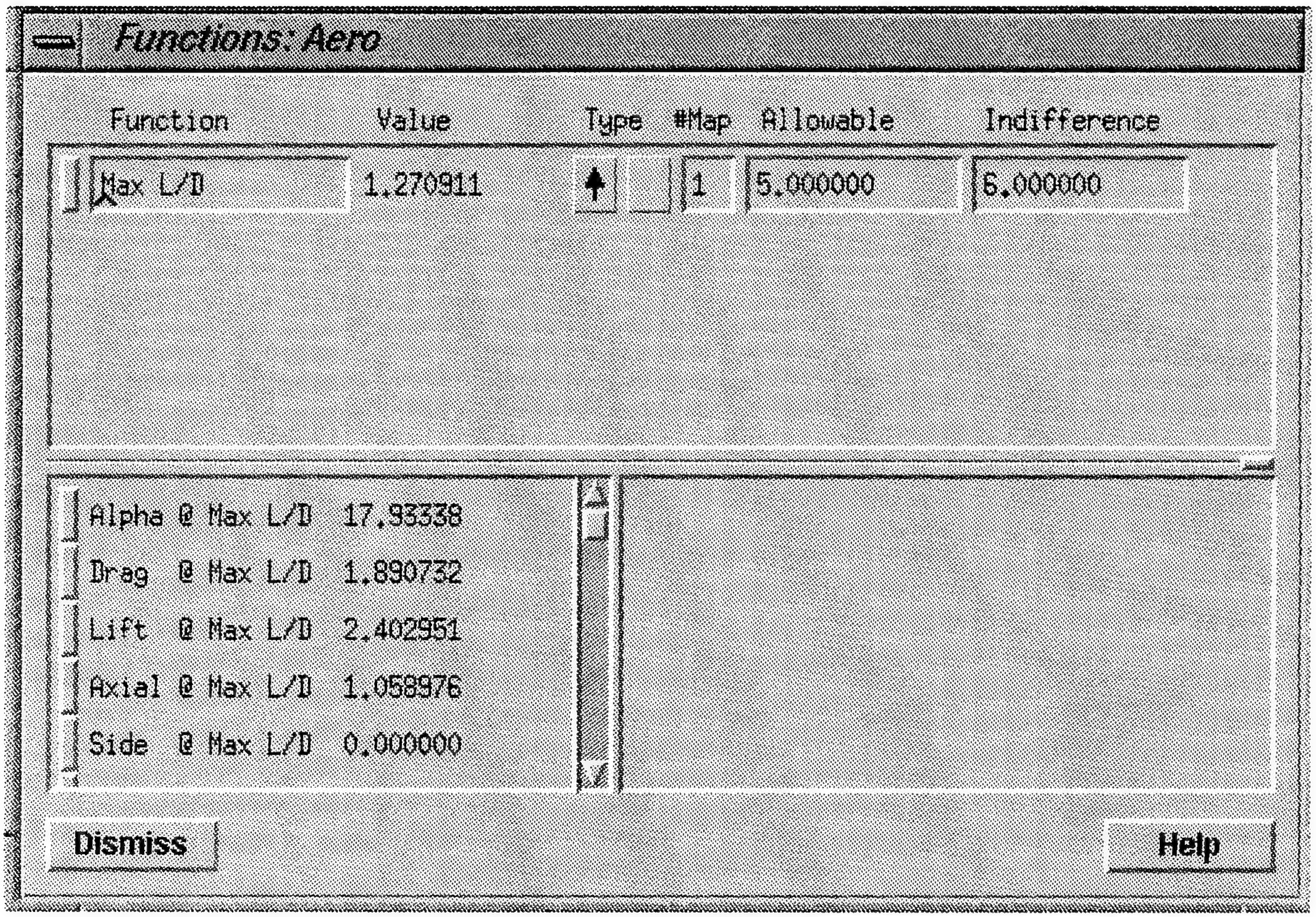
Step 4: The New File window initializes the variables and functions. Save the initial analysis as **imminit**.



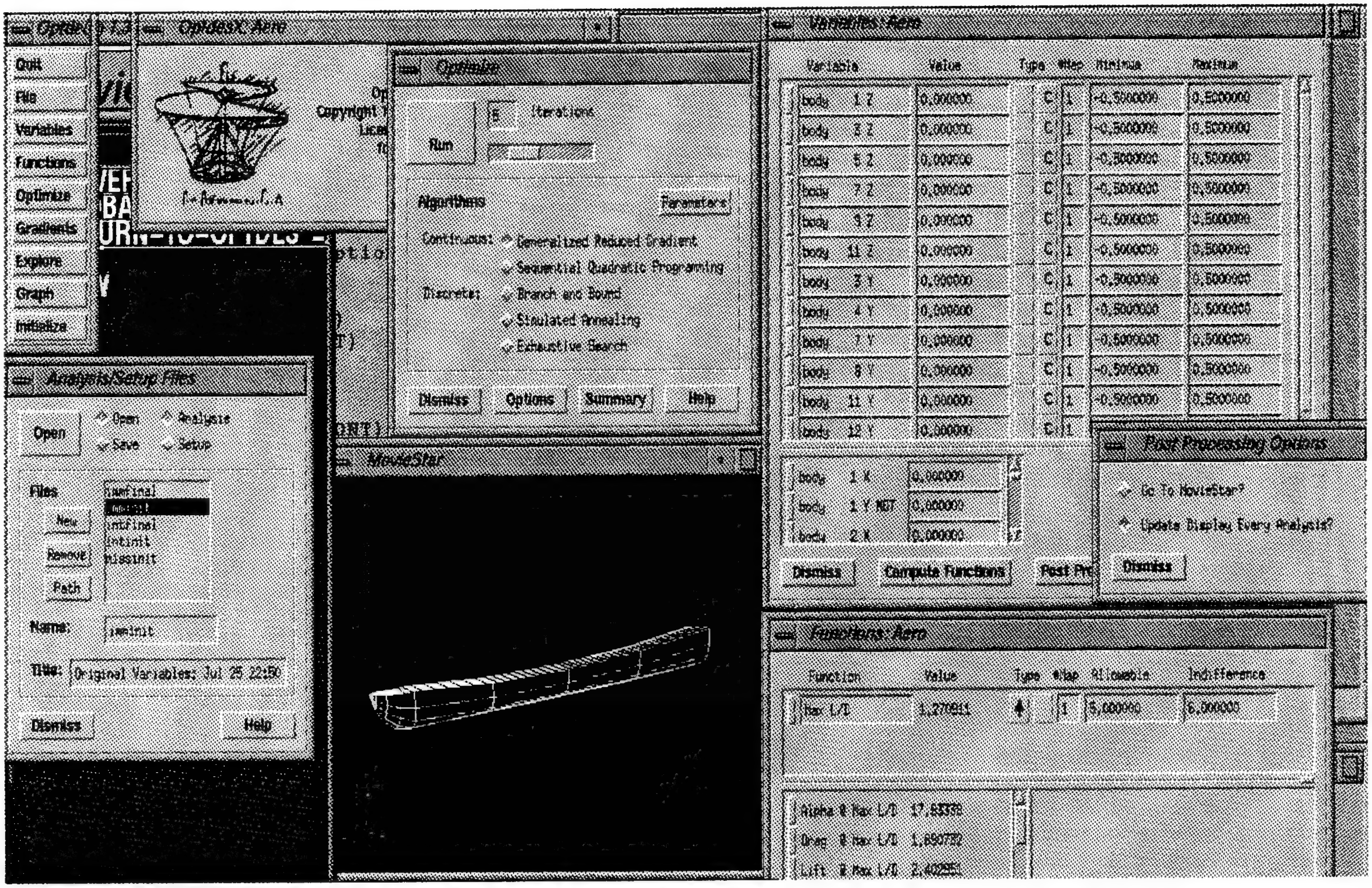
Step 5: Define the Variables window as follows.



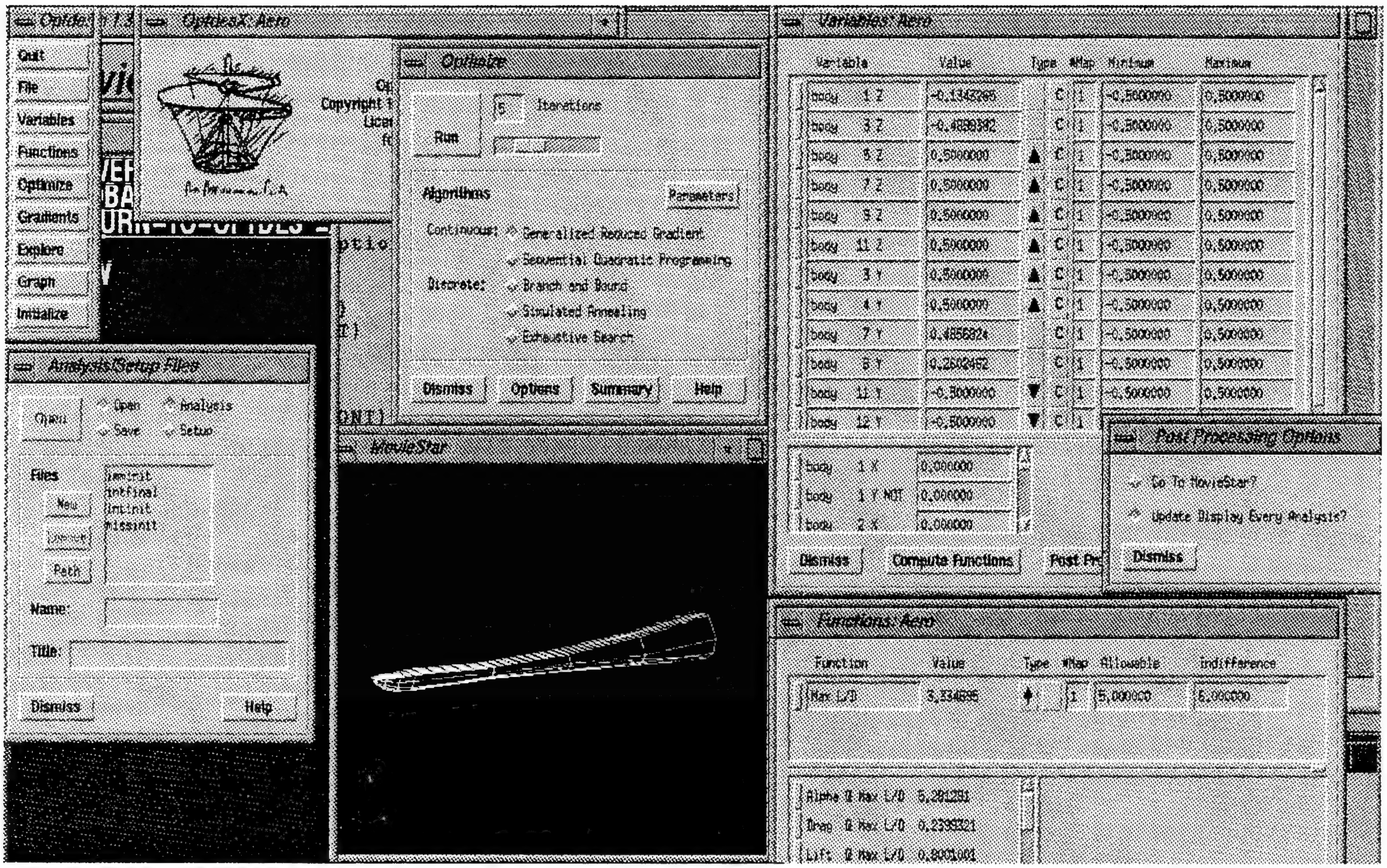
Step 6: Define the Function Window as follows.



Step 7: The image here is the the complete screen at the initial state just before we Optimize.



Step 8: The following shows the complete screen showing the final design ("optimal") solution. We see that throughout the complete design evolution, the top surface remained undeformed and the shape did change to increase or maximize the L/D.



THE END.

Appendix B

TrajView User's Manual And Tutorial

Trajectory Viewer Tutorial

Introduction:

The Trajectory Viewer (TrajView) is used to perform real time 3–D animations of CADAC simulations. Use, duplication or disclosure is subject to restrictions stated in Contract # F08630–91–C–0052 with APTEK, Inc.

Capabilities:

The following capabilities are available in TrajView. All of these are under user control.

The simulations can be performed using a flat earth grid or a globe model.

Animations can be performed at real time, fast forward, reverse, fast reverse and slow motion (forward and reverse).

Flat earth grid size can be set by user.

Multiple bodies or multiple trajectories (single body) can be animated at the same time.

Sensor pyramids can be animated. The user has the option of moving the camera inside the sensor to "see" what the sensor sees.

INS error ellipses can be displayed. The resolution and magnification of the ellipses is user controlled.

Velocity vector can be displayed for a particular body.

Flight and ground traces can be performed.

Different models can be used for each body. Current models available are the F16, F15, Missile, Tank, Bridge, X, and RentryVehicle.

Altitudes can be scaled in order to increase visualization.

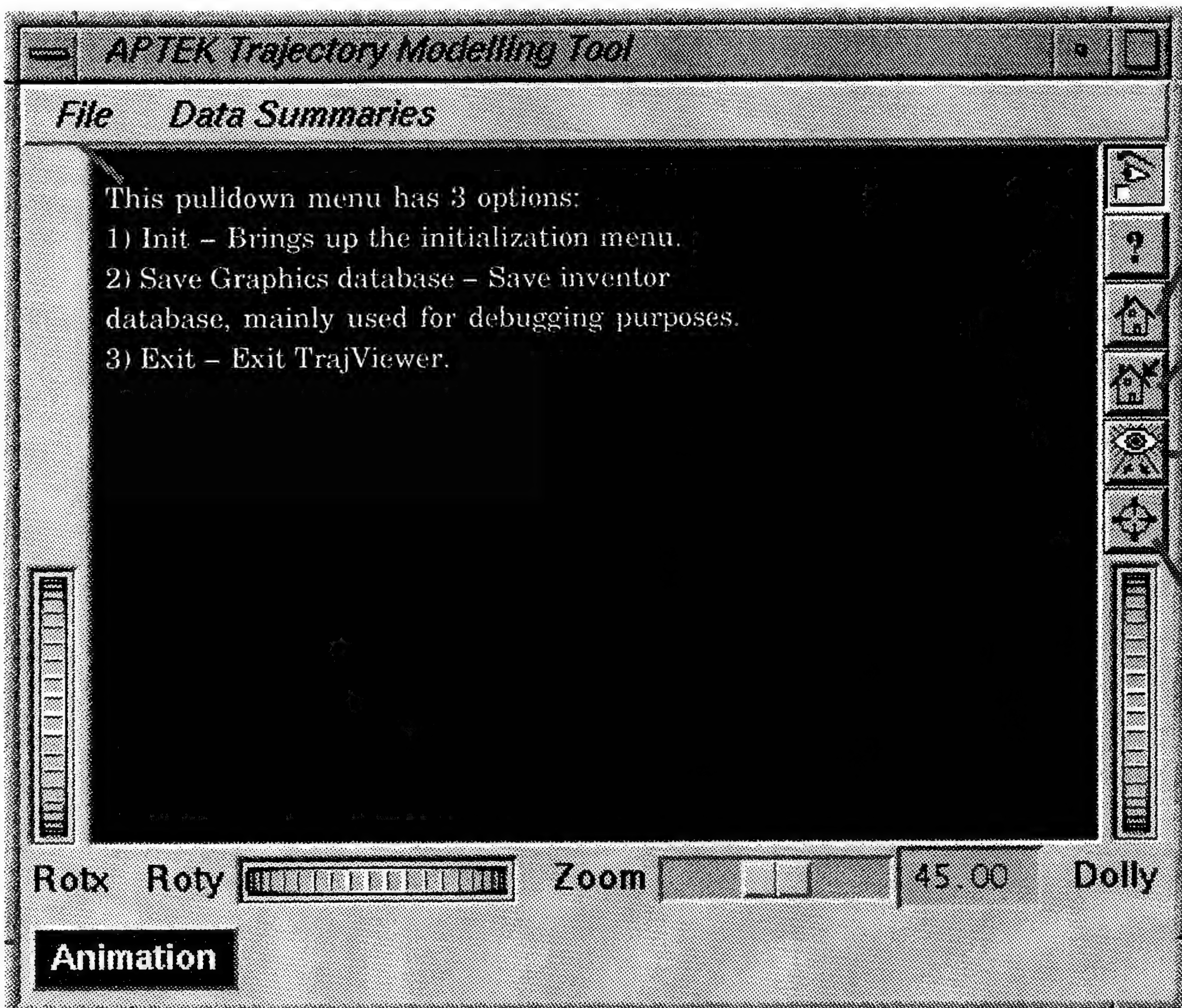
X/Y Plots can be generated.

CADAC variables can be displayed "real time" with sliders during the animation.

Example 1 – Flat Earth:

The first example will be using a flat earth model for the animation. The CADAC file is LSF.3_APTEK JDAM3- IIR. First we will set up the animation, then APPLY the data which builds the 3-D models, then perform the animation using a VCR type display.

1. In a console window, type TrajView and two windows will appear. The first window is the 3-D animation window. In the black area below is where the animation will be performed. The slider wheels (Rotx, Roty, Dolly) allow you to move the camera to any orientation. Zoom allows you to zoom into the picture.



Moves camera back to HOME.

Creates a new HOME.

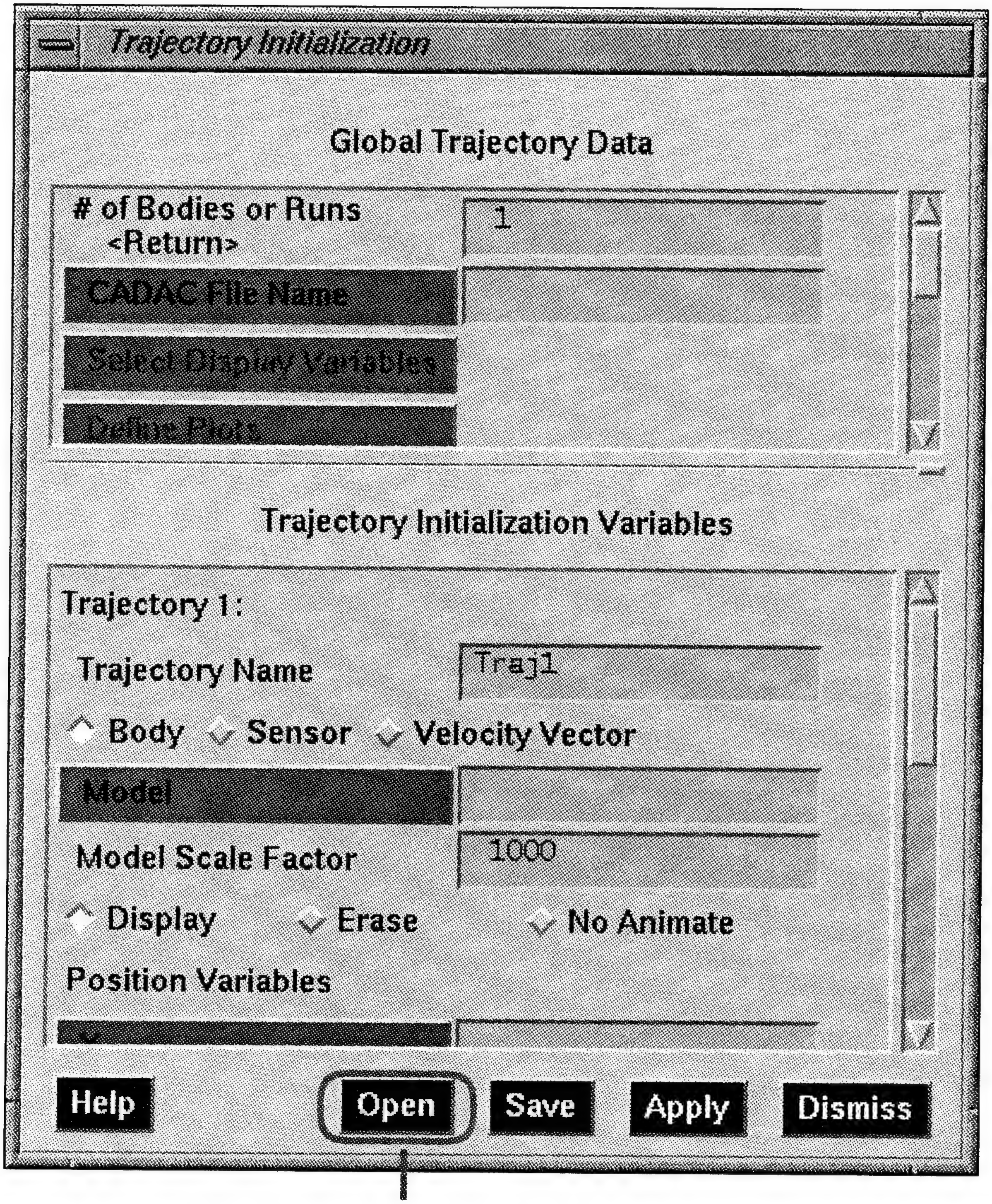
Centers objects on the screen.

Click to define a surface normal to the screen.

Press to bring up VCR display for controlling animation.

Example 1 – Flat Earth:

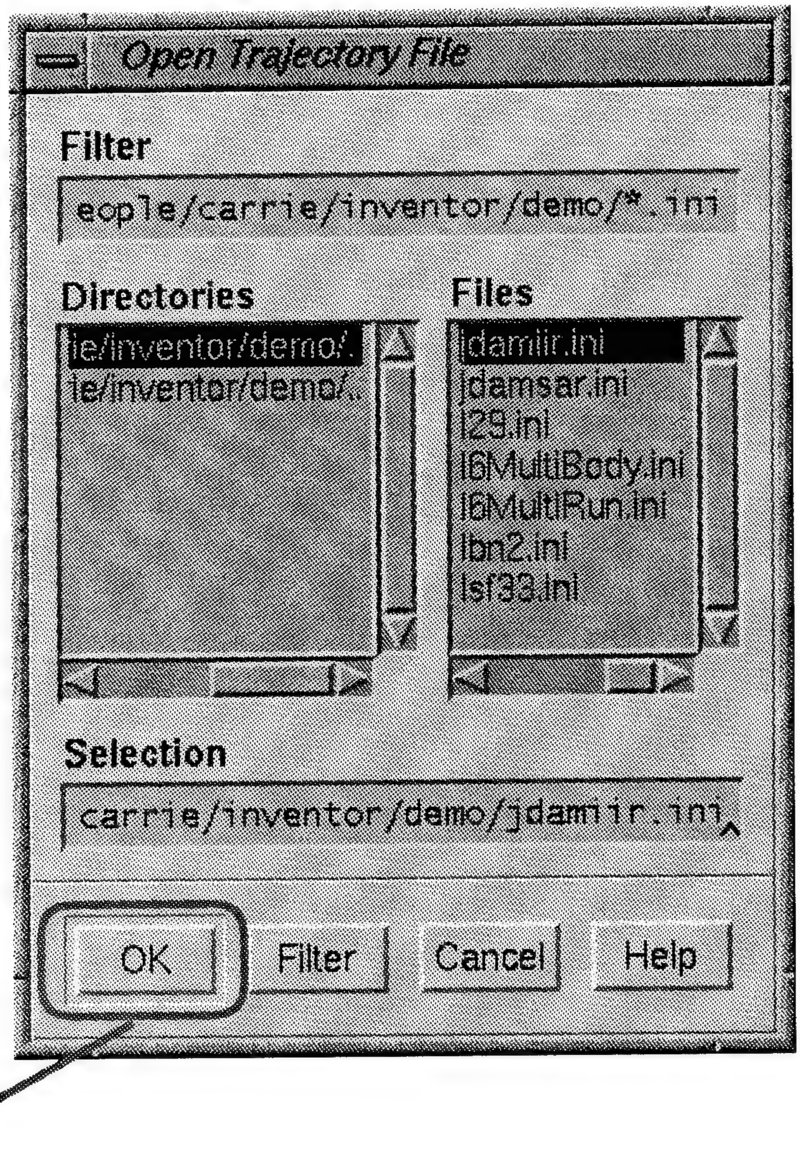
2. The second window to appear is the Trajectory Initialization menu. This menu needs to be filled out by the user in order to define the animation. All data entered in this menu can be SAVEd and OPENed. For this tutorial, we will OPEN an initialization file that will automatically fill in the contents of this menu.



OPEN button to Load Initialization Menu

Example 1 - Flat Earth:

3. Press the OPEN button, this will bring up a file selection menu. Click on the file jdamiir.ini.



Once you have selected the correct file, Click on the OK button. This will then load the Trajectory initialization window with the data specified in this file.

Example 1 - Flat Earth:

4. Two additional menus will pop up, one for defining the Sensor and the other for defining the velocity vector and the error ellipse display. These will be explained later after we first explain the initialization menu.

   The trajectory initialization menu is used to set up and initialize the trajectory animation. There are two parts to this menu. The GLOBAL TRAJECTORY DATA and the TRAJECTORY INITIALIZATION.

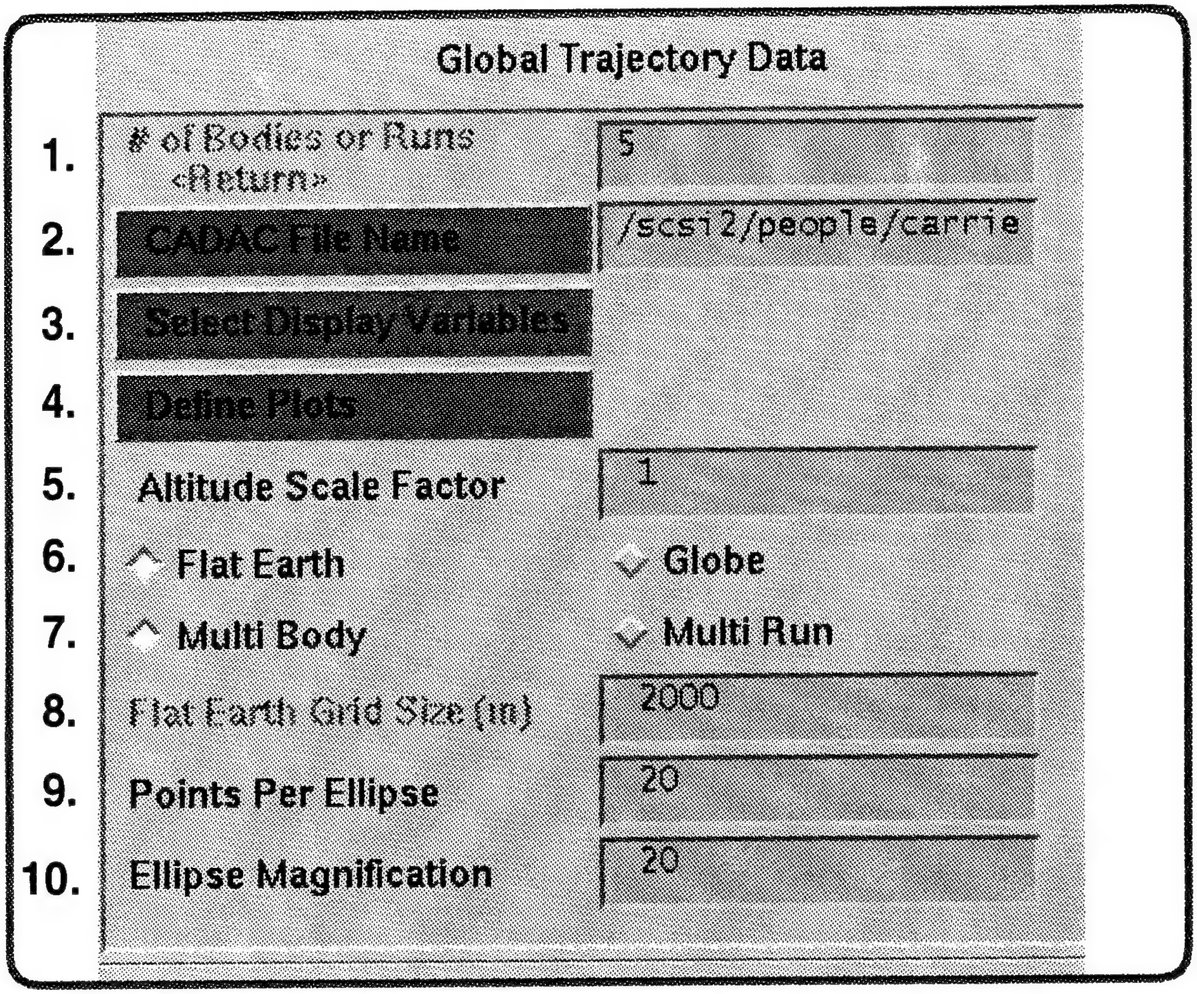
Global Trajectory Data:

The Global trajectory data contains information that applies to all of the trajectories in the animation. For each Body or Run specified in the Global section a trajectory initialization window is built in the TRAJECTORY INITIALIZATION section.

Trajectory Initialization Variables:

This section of the menu contains trajectory specific data such as X, Y, Z variables, what model to use, what type of model, etc.

The next three pages will explain what has been loaded into the initialization menu. You are not required to type anything in for this tutorial.



1. Number of Bodies (for a MultiBody run) or Number of Runs (for a single Body run). Can only have one or the other, not both.
2. Pathname for CADAC file. Press Green button for File Selection Window.
3. Press to select variables to display during animation.
4. Press to define Plots. Must APPLY data (button found on bottom of INIT menu) first.
5. Scale factor for altitude.
6. Select Flat Earth or Globe model. Flat Earth's translations are X, Y, Z and Globe's translations are Latitude, Longitude, Altitude.
7. Select Multi Body or Multi Run. Multi Body contains multiple bodies (shooter, target, sensor, etc.). While a MultiRun contains a single body with multiple CADAC runs.
8. Flat Earth Grid size (in Meters).
9. Number of points used to define an Ellipse. Ellipses are used to display INS errors.
10. Magnification of ellipses used to display INS errors.

Trajectory Number

Name of Trajectory

Trajectory Type. Body is used for shooter, missile or target. Sensor and velocity vectors are special types and are considered separate trajectories.

Type of model used for this trajectory. Press MODEL to get a list of models. Will search current working directory for model then will search Model Directory.

Scale factor of model. All models are in meters and actual size.

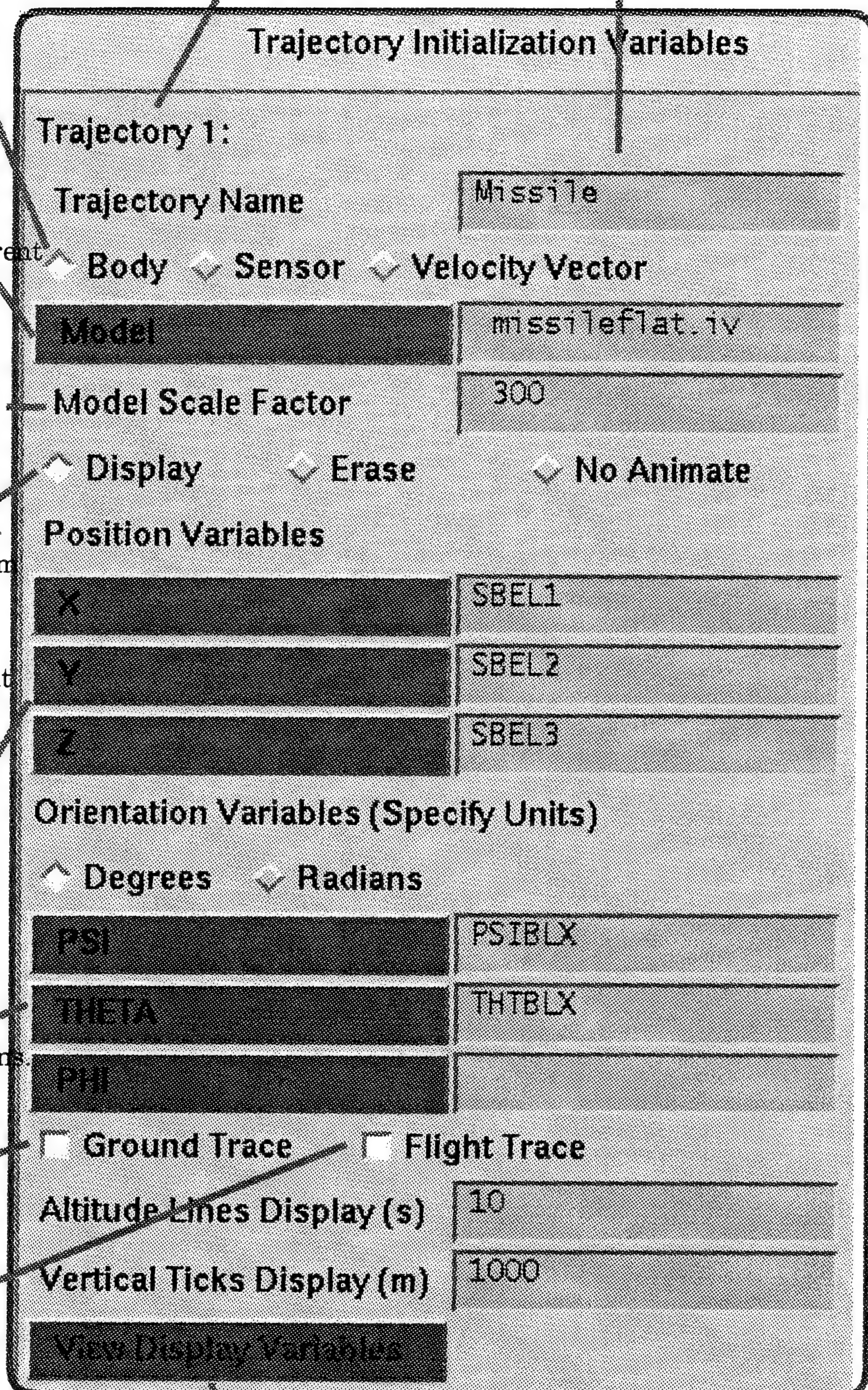
Display means display trajectory. Erase means erase trajectory from display and do not animate. No animate leaves trajectory at current position on the display but does not animate it.

X, Y, Z position variables for flat earth. For Globe, Latitude, longitude, altitude. Press button for list of variable names.

Orientation variables for model. Specify whether degrees or radians. Press button for list of variable names.

Turn on/off ground trace for this trajectory.

Turn on/off flight trace for this trajectory. If flight trace, specify how often vertical stringers will be displayed (secs) and how often vertical ticks will be displayed (meters). Cannot be changed once APPLY pressed.

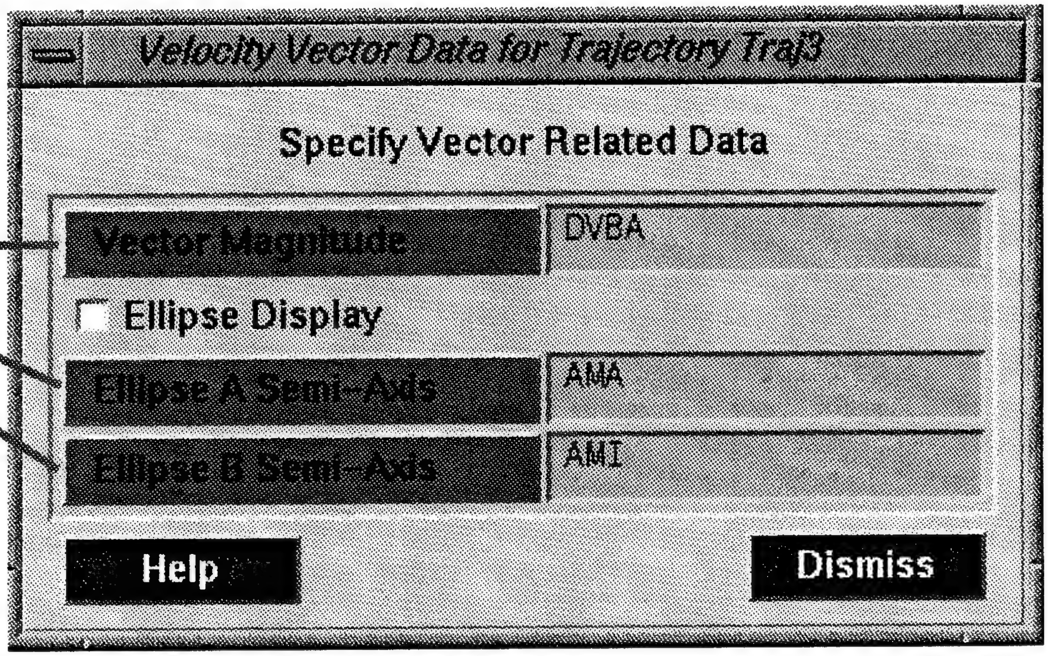


After APPLY is pressed and Display variables have been selected in the GLOBAL section, can press View Display Variables to bring up slider scales to display variables during the animation.

Example 1 - Flat Earth:

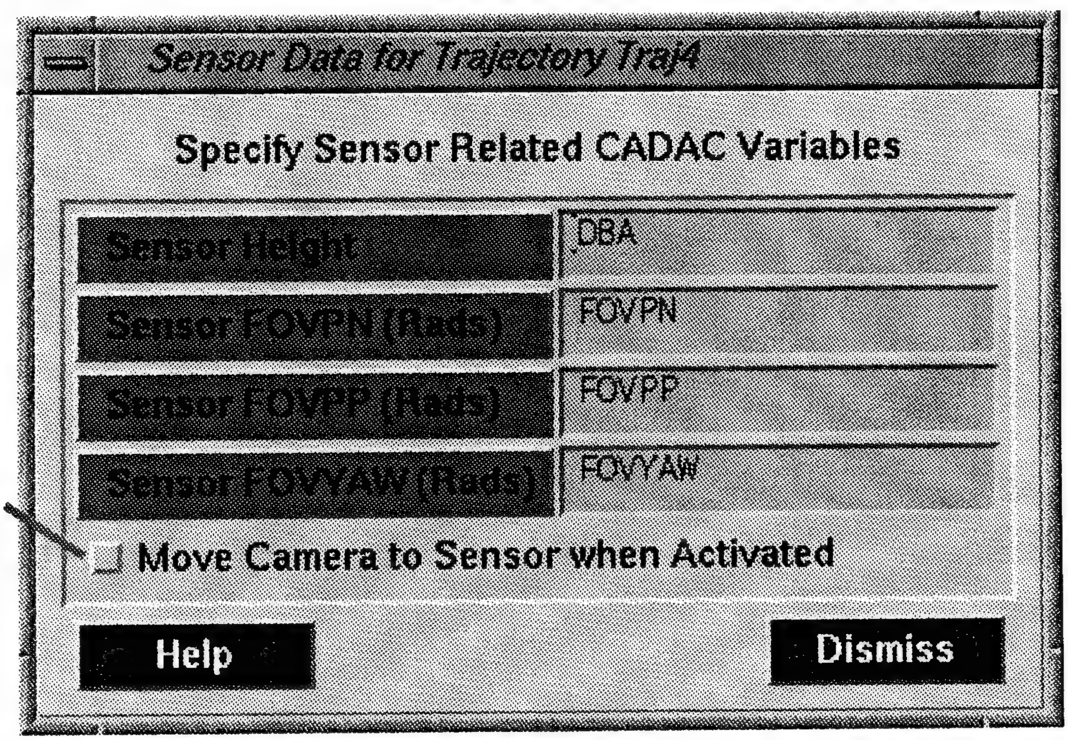
A velocity vector was defined for the third trajectory (Traj3). The following menu was also initialized to define the vector magnitude and to turn on the ellipse display, with major and minor axis for the ellipse. The ellipses can only be defined in conjunction with a velocity vector, since they are defined to be normal to the velocity vector.

Press Green button to get a list of CADAC variables. Double clicking on variable in list will automatically place the variable in the appropriate field.



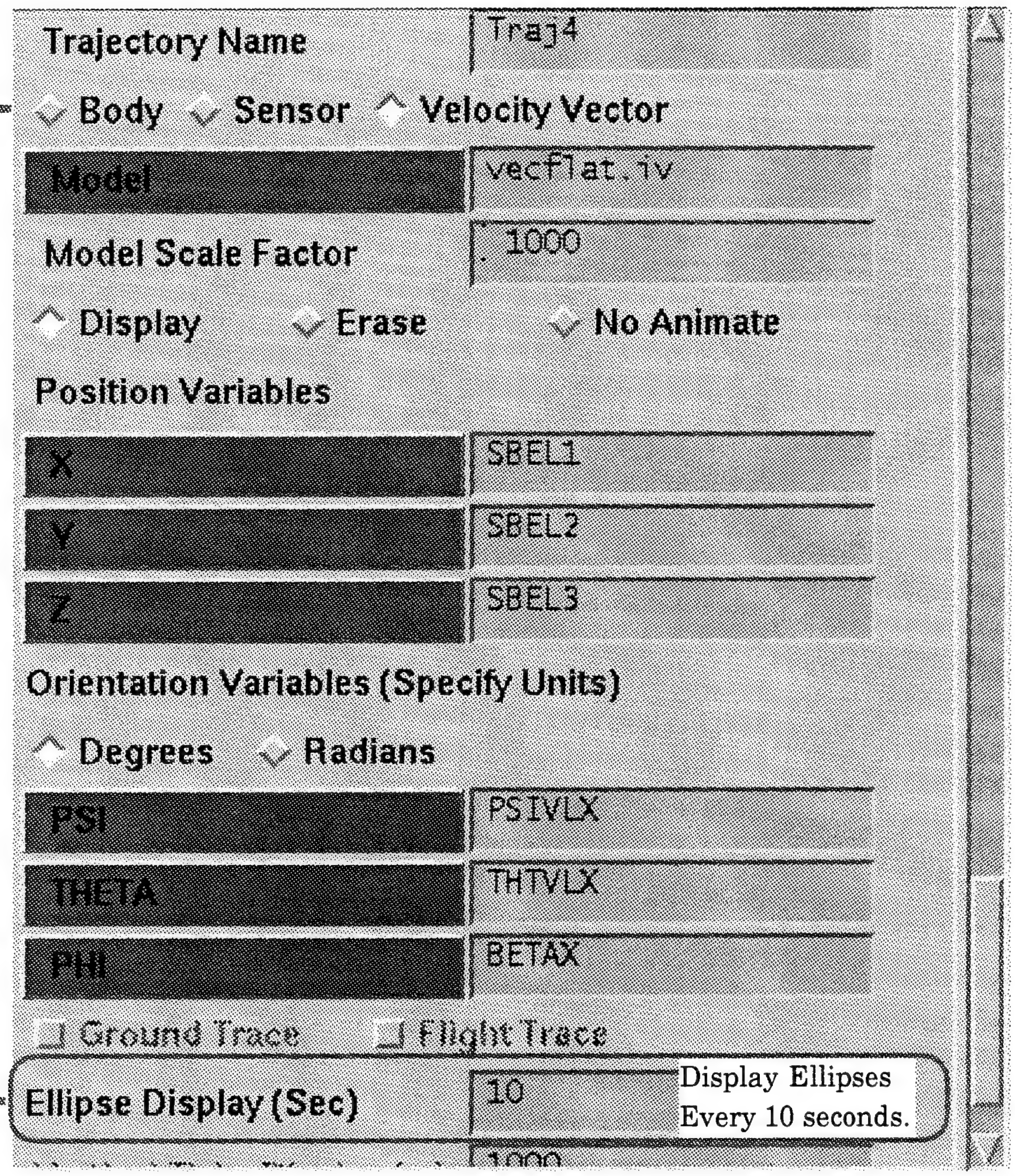
A sensor was also defined. The following menu defines the sensor.

Click on this to "See" what the sensor sees. Will only be in effect when sensor is activated (Sensor Height > 0).



Example 1 - Flat Earth:

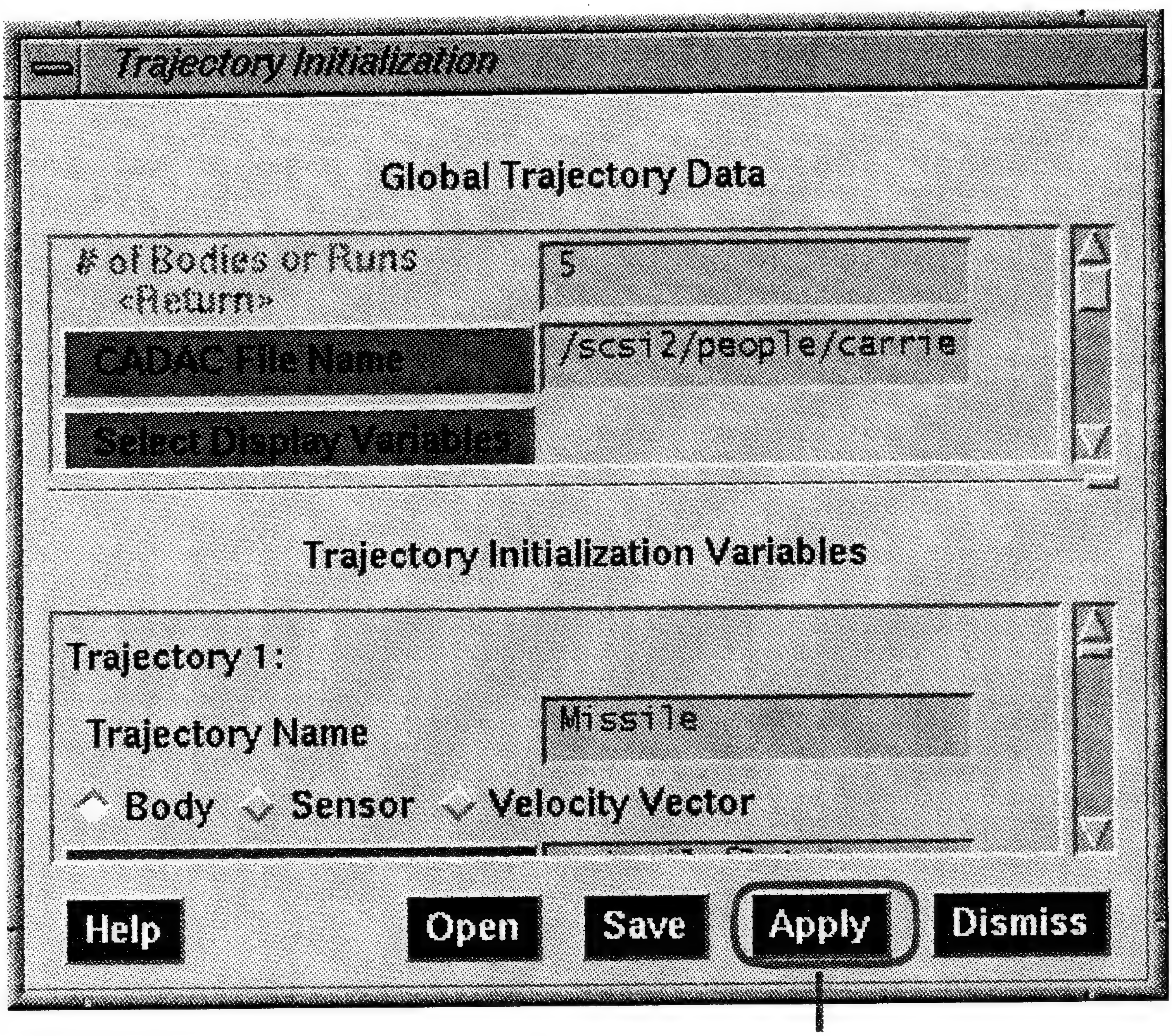
When defining the ellipses to display the INS errors, you can specify how often you want the ellipses displayed. This is done in the INIT menu for this particular trajectory. When Trajectory type is of type Vector, the altitude display is renamed Ellipse Display on the initialization menu. Here you specify how often you want the ellipses created.



Altitude display renamed Ellipse Display when trajectory of type Velocity Vector.

Example 1 - Flat Earth:

5. Once the data is entered correctly, Press the APPLY button to build the 3–D scene. The 3–D scene will then be loaded in the animation window.



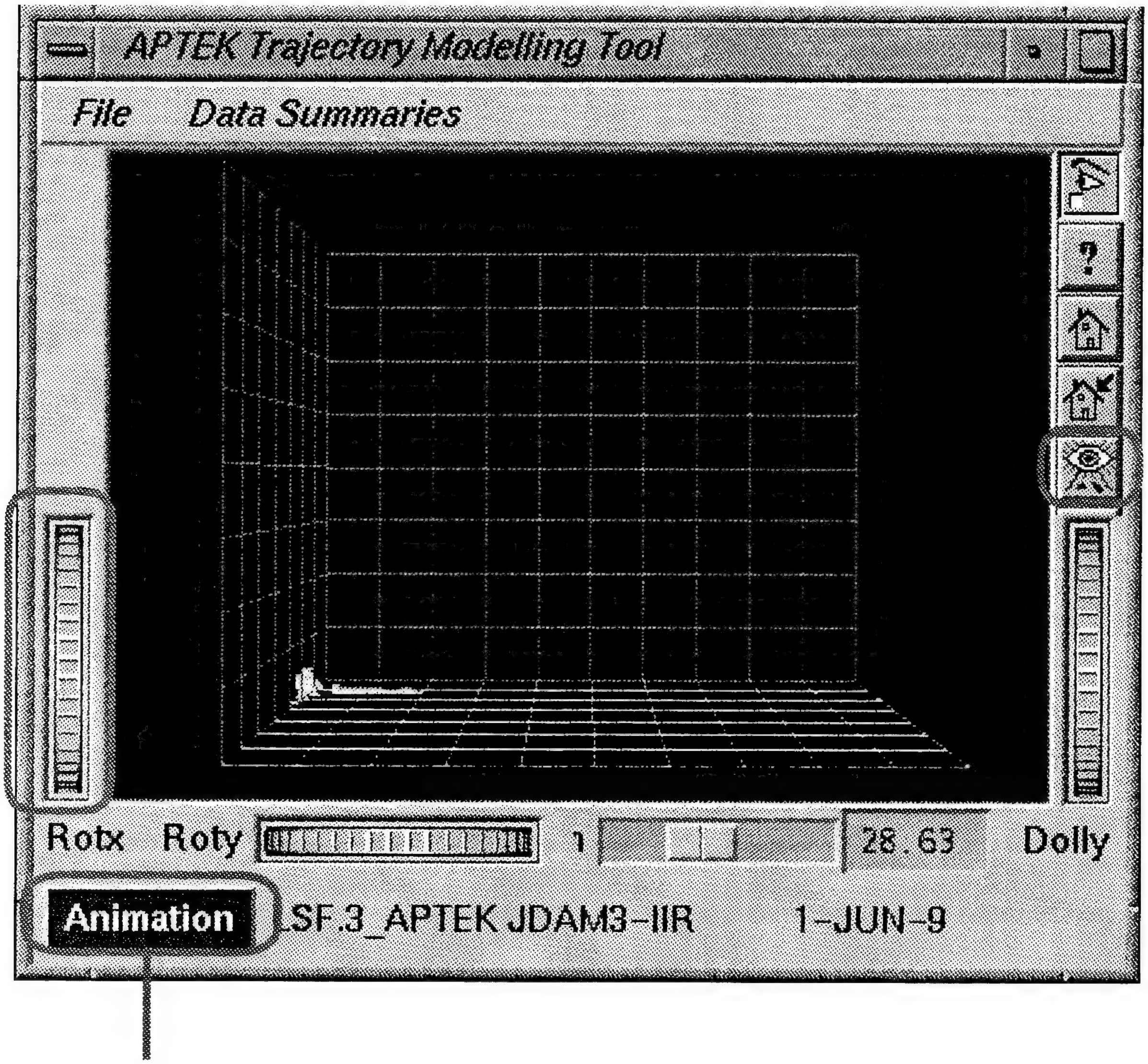
Press APPLY to load CADAC data and build a 3–D scene from it.

Note: After pressing APPLY, a CADAC summary window will appear. This simply summarizes the CADAC data and can be dismissed.

Example 1 - Flat Earth:

6. The scene will be loaded looking down the Z axis. Therefore, we are looking at the bottom of the flat earth. To get the flat earth in its proper orientation 1) first click on the Center button (Eye button on right side), 2) then use the Rotx slider to rotate the flat earth 90 degrees.

2.
Rotate Flat Earth 90 about the X to get in proper orientation.

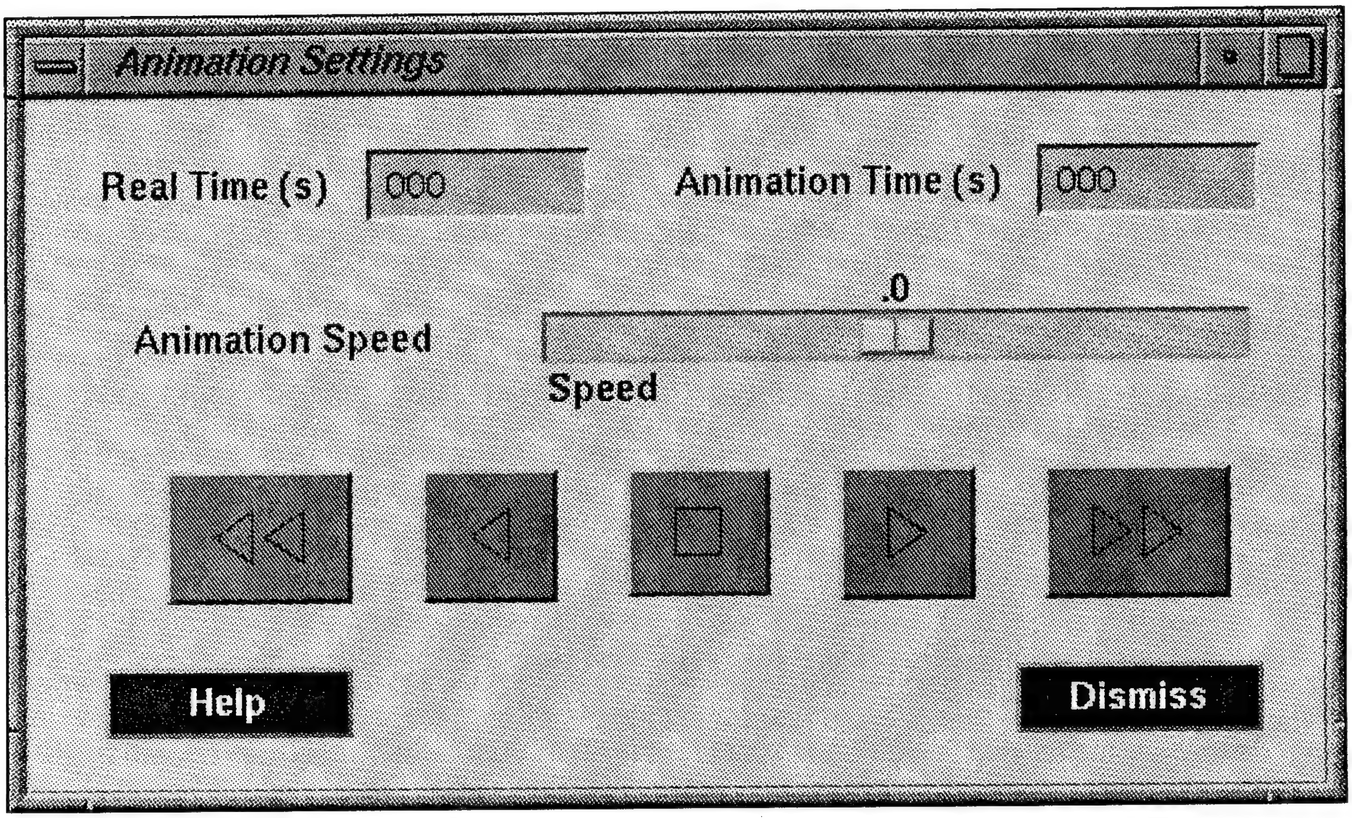


1.
Click on Eye button to CENTER flat earth.

7. Press the Animation button to bring up VCR controls for starting and stopping the Animation.

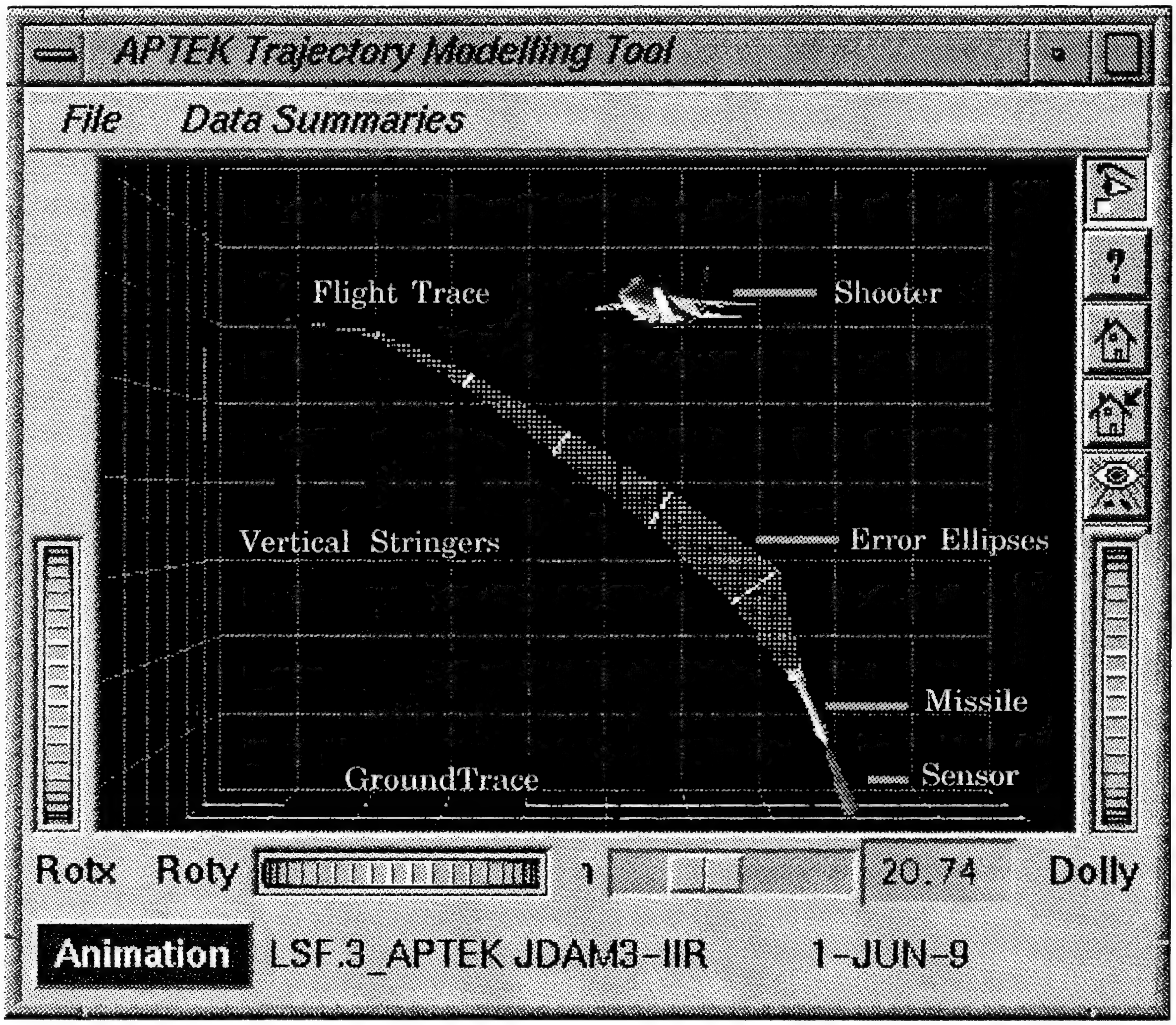
Example 1 - Flat Earth:

8. Start the Animation! You can move forward or backwards by pressing the buttons or by using the slider. The resulting animation will be displayed in the animation window.



Example 1 - Flat Earth:

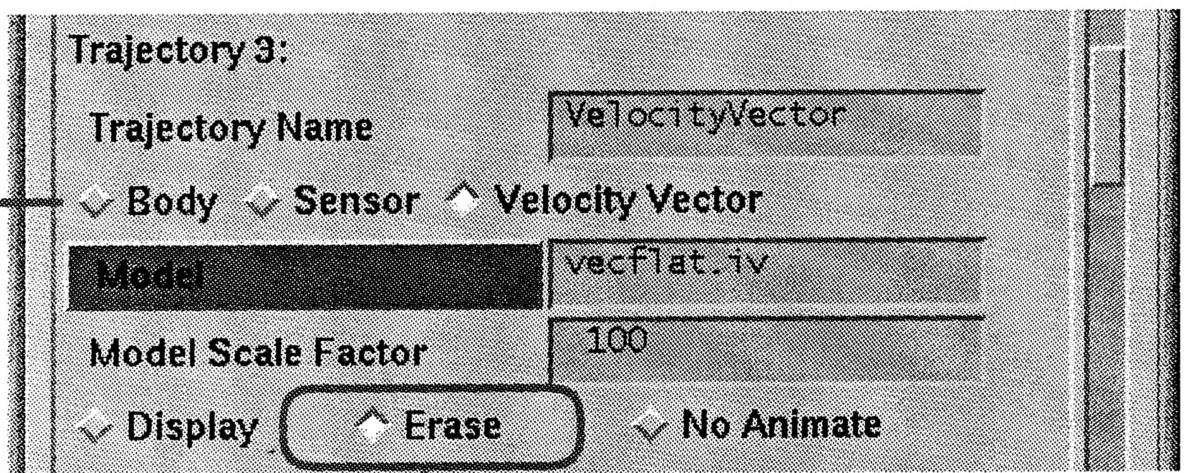
9. A snapshot from the animation is displayed below. Try moving forward and reverse and turning on and off various trajectories (via the Initialization menu). Also try rotating the 3-D scene and Zooming in and out.



Example 1 - Flat Earth:

10. On the Sensor definition menu, click on the Move Camera to Sensor option. If you have dismisses this menu, you can get it back by moving to the 4th trajectory on the Initialization menu and toggling the Trajectory type between body and sensor. Once you press Sensor, the menu will appear.

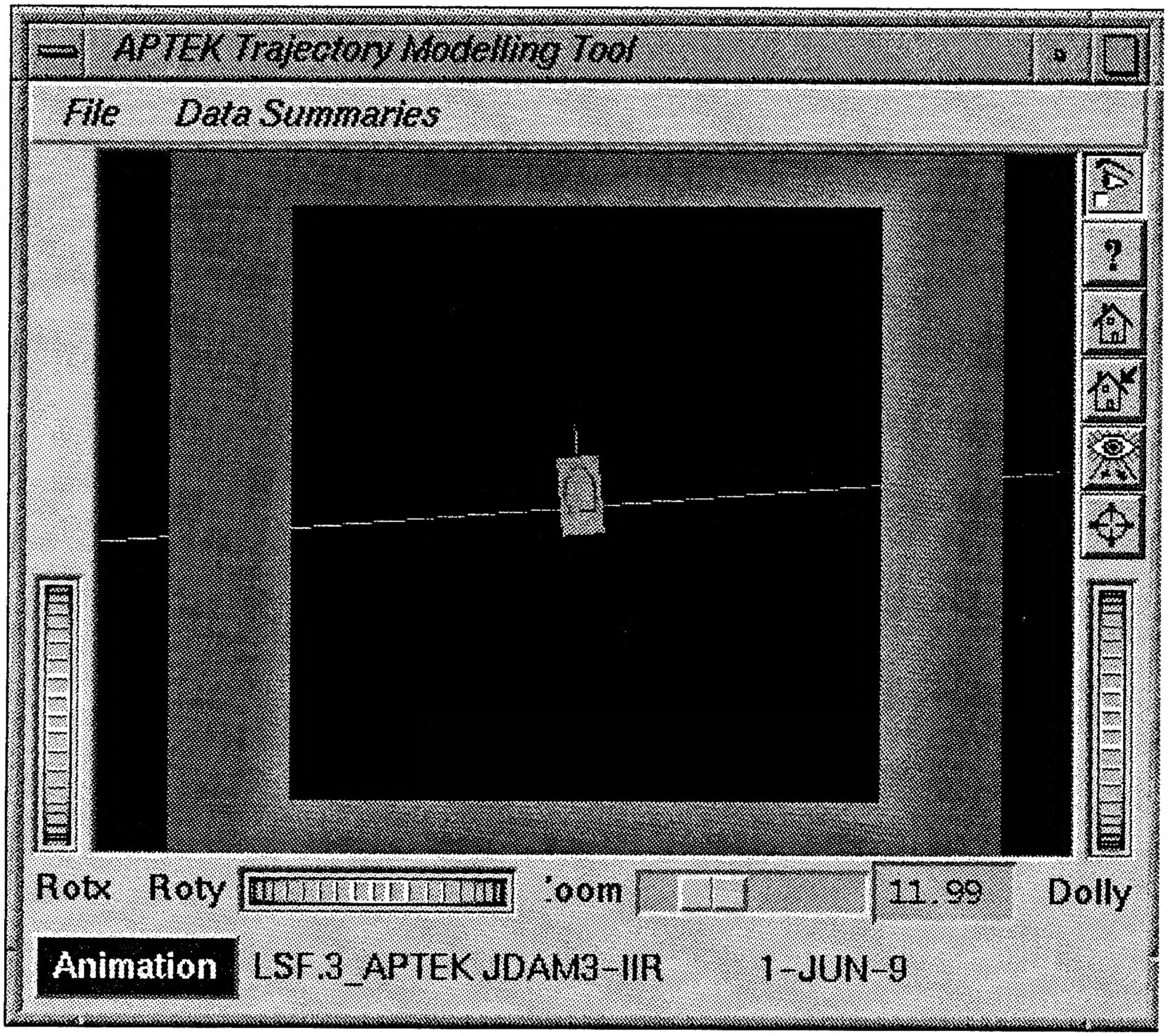
Rewind the animation and start again with the camera to sensor option selected. When the sensor becomes active, the camera will be activated so that it appears that you are riding on the missile, looking down the sensor field of view. However, in order to see down the sensor field of view you must ERASE the Velocity Vector



Erase Velocity vector before moving camera to sensor!

Example 1 - Flat Earth:

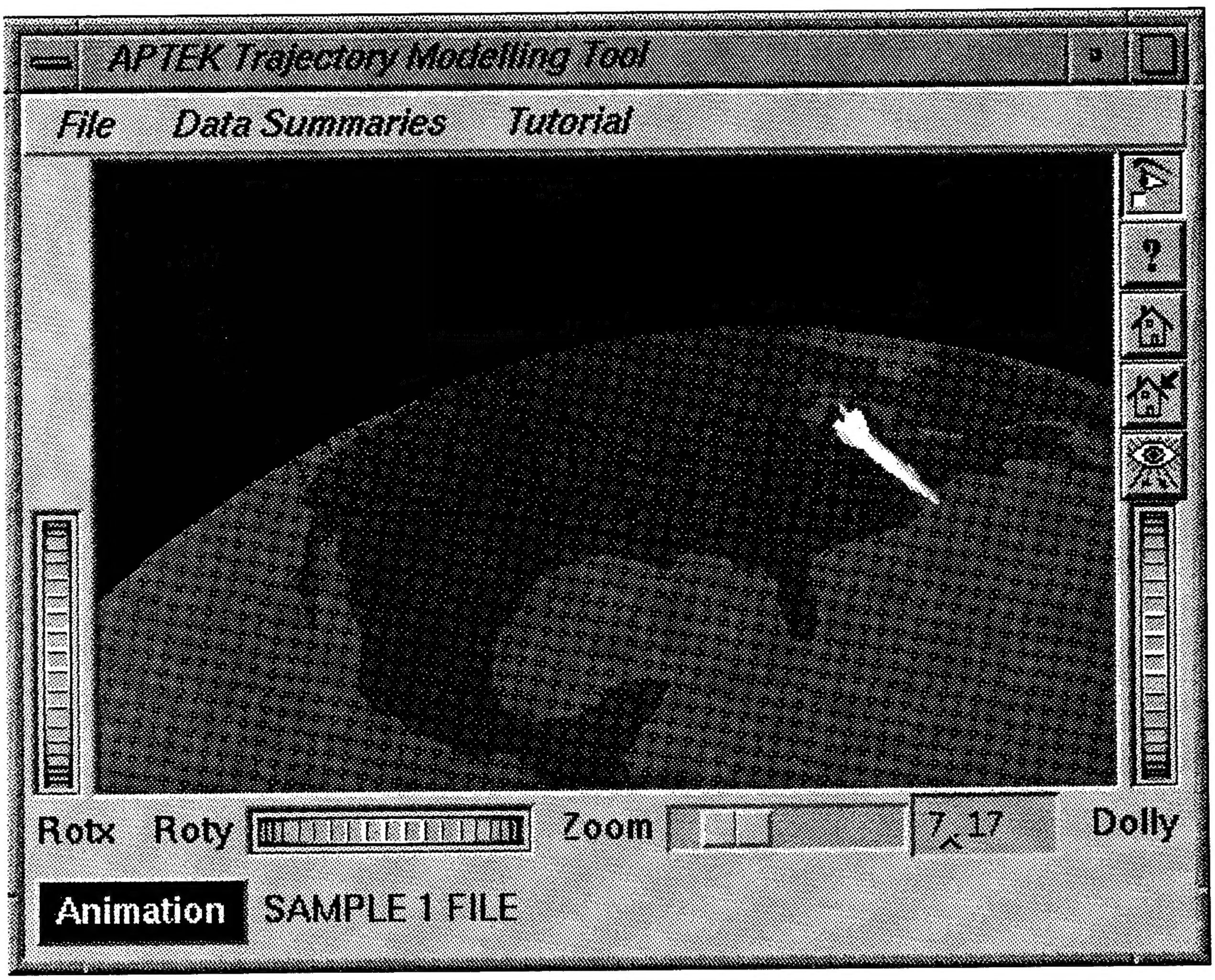
The result of moving the camera to the sensor is shown below. The sensor is in magenta and the target is the tank below.



Example 2 - Globe Model:

The second example will be using a globe model for the animation. The CADAC file is SAMPLE1 FILE. The globe model will run much slower than the flat earth due to the complexity of the model. The more complex the model, the slower the animation.

1. Type TrajView in Console window.
2. Press OPEN on init menu. Load file globe.ini. Note that you use different models for the globe versus the flat earth (i.e. RVFlat.iv vs. RVGlobe.iv).
3. Press APPLY button.
4. The camera is originally in the center of the Earth. Press the Eye CENTER right hand side button to center model in picture.
5. Press the animation button and begin the animation. You may want to zoom in and rotate the camera to get a better view.



DISTRIBUTION LIST

ASC/XRED, (Bldg 11A) 1970 Third Street, Suite 7 Wright-Patterson AFB, OH 45433-7214 Attn: Dave Hammond	1
WL/FIGC Wright-Patterson AFB, OH 45433-6523 Attn: Jim Simon	1
WL/MNAV Eglin AFB, FL 32543-5000 Attn: Fred Davis	1
WL/MNPM (STINFO Facility) Eglin AFB, FL 32543-5000	1
DTIC Attn: DTIC/DDAC Cameron Station Alexandria, VA 22304-6145	2
APTEK, Inc. 1257 Lake Plaza Drive Colorado Springs, CO 80906-3578	1